TU MICRO

COMMODORE

N.º 12 · SEGUNDA EPOCA

375 PTAS · (IVA INCLUIDO)



RUTINAS KERNAL

SECRETOS DEL DATASSETTE

DIGITALIZACION DE IMAGENES

CON COMMODORE



CIMEX ELECTRONICA — Nº 1 EN APLICACIONES Y UTILIDADES PARA **Commodore**

SUPERCONTA 64

Programa realizado en código máquina, lo que implica mayor rapidez y ocupa menos memoria.
Tiene una capacidad de 475 cuentas, 5300 apuntes de disco y 305 apuntes por día. El límite de 475 cuentas se debe a que están siempre en memoria y se puede acceder a ellas instantáneamente.
Se pueden introducir asientos en cualquier fecha, también atrasadas, él mismo genera el número de asiento y además visualiza el número de línea, descuadre del asiento y título de cuenta, también se puede dar de baja, el programa actualiza automáticamente todo, se puede introducir un asiento en un solo apunte.
Los listados se pueden hacer en cualquier fecha y las veces que se desee:

disco. 19.900

PERSPECTIVAS V2

Procesador de imágenes de figuras volumétricas, que permite obtener en gráficos de alta resolución, perspectivas cónicas, axonométricas, planta y alzado de cualquier figura previamente definida, a partir de las coordenadas de sus vértices y de la disposición de sus aristas. Permite entrar cualquier figura (edificios, piezas mecánicas, objetos de diseño, etc.). Es de gran interés para estudiantes y profesionales de la arquitectura, ingeniería, diseño y también para aplicaciones didácticas en el campo de la geometría y dibujo.

PERSPECTIVAS V2

AHORA CON SALIDA POR IMPRESORA

PERSPECTIVAS V2 (C) Cimex Electrónica 1986
Pabellón barcelona (mies van der rohe)
Perspectiva cónica (50, 15, 20)

(d) 5.000 (c) 4.500.






5 1/4 CENTRO REFORZADO CALIDAD GARANTIZADA.

SS/DD ESPECIAL COMMODORE, APPLE, ATARI (10 unidades) 1.750-

DS/DD. ESPECIAL PC Y COMPATIBLES (10 unidades) 2.050-

SOFTWARE PARA COMMODORE 64

COMPILADOR	(d) 5.000
CONTABILIDAD PERSONAL	(d) 3.000 (c) 2.500
EDITOR DE ETIQUETAS	(d) 6.000 (c) 2.500
SIMULADOR DE SPECTRUM	(c) 2.500
BASE DE DATOS	(d) 8.000 (c) 3.500
MUSIC 64	(d) 3.500 (c) 3.000
PERSPECTIVAS	(d) 5.000 (c) 4.500
GESTION DE STOCKS	(d) 10.000
EDITOR DE RECIBOS	(d) 15.000
PROTEXT	(d) 7.950
AYUDA AL PROGRAMADOR	(d) 3.000 (c) 2.500

THE FINAL CARTRIDGE

UN CARTUCHO INCREIBLE... LAS UTILIDADES

TURBO DISCO: carga y salva los programas 6 veces mas rapido. No borra la pantalla.
TURBO CINTA: carga y salva 10 veces mas rapido. utiliza los comandos normales LOAD SAVE.
INTERFACE CENTRONICS: permite utilizar las impresoras de tipo paralelo. Imprime los graficos Commodore asi como los codigos de control en los listados.
VOLCADOS DE PANTALLA: de alta resolución, texto en pagina de ancho. Permite imprimir pantallas de juegos o los dibujos realizados con **DOODLE, KOALA PAD, PRINT SHOP** etc. Busca automaticamente la dirección de memoria. Funciona con impresoras Commodore, Compatibles y Centronics.
24 K. MAS DESDE EL BASIC: 2 nuevos comandos "Memory Read" y "Memory Write" mueven 192 bytes muy rapidamente en cualquier sitio de los 64K de RAM del 64. Lo que permite guardar las variables y las cadenas de caracteres debajo de la ROM.
COMANDOS DEL BASIC 4.0. Dload, Dsave, Dappend, etc.
AYUDA A LA PROGRAMACION: como Auto, Renum (incluso GOTO y GOSUB) Find, Help, Old.
TECLAS DE FUNCION PROGRAMADAS: Run, Dsave, Directorio, Comandos de disco, Dload, List, Old, Monitor. Por supuesto, puede volver a programarlas.
POTENTE MONITOR DE CODIGO MAQUINA: Scroll hacia arriba/abajo, Bankswitching (para levantar la ROM), etc. NO RESIDE EN MEMORIA. Se puede llamar en cualquier momento con cualquier programa en memoria.
MONITOR DE DISCO: para leer o escribir un sector determinado del disco, cambiarlo de sitio, leer la RAM del disco, etc.
RESET: resetea TODOS los programas.
VARIOS: el comando TYPE permite utilizar ordenador e impresora como si se tratase de una maquina de escribir electronica. Puede incluir en sus programas los POKE y los SYS en hexadecimal.
GAME KILLER: Anula la deteccion de colision de sprites, para ir hasta el final de los juegos.

EL FREEZER

Apretando el pulsador del Freezer tomara el control de su ordenador, congelando el programa en memoria. Con la ayuda de menus muy comodos podra:
— Hacer VOLCADOS DE PANTALLA alta o baja resolucion (por ejemplo las pantallas de sus juegos preferidos). Congele cualquier programa en el momento que desee y vuelque la pantalla sobre el papel.
— Cambiar los COLORES en sus juegos.
— Llamar en cualquier momento y con cualquier programa ejecutandose el MONITOR de codigo maquina.
— Hacer COPIAS DE SEGURIDAD de sus programas. El Freezer le permitira con programas protegidos cualquiera que sea el sistema de carga utilizado (turbos, verificacion de errores, entre-pistas, etc.)

- Hacer COPIAS DE CINTA A CINTA
- Hacer COPIAS DE CINTA A DISCO
- Hacer COPIAS DE DISCO A CINTA
- Hacer COPIAS DE DISCO A DISCO

solo pulsando una tecla. El proceso de copia es totalmente automatico y el utilizador no necesita tener conocimientos algunos de programacion. La copia facilitada por el Freezer consta solo de 2 PARTES (el cargador y el programa propiamente dicho) y se puede cargar SIN EL CARTUCHO a velocidad turbo.
El Freezer de THE FINAL CARTRIDGE 2 es mas potente, mas rapido y mas comodo de usar que la gran mayoria de los productos especializados ingleses o americanos.

¡Hasta el precio es increíble!



CINTA C-10 (10 unidades)	890
CINTA C-20 (10 unidades)	990
CABLE CENTRONICS C-64	3.450
FUNDA C-64 y VIC-20	850
JOYSTICK QUICKSHOT II	1.390
JOYSTICK QUICKSHOT I	990
JOYSTICK QUICKSHOT II + (con microrruptores)	2.595
KIT AJUSTE DATASSETTE	2.395
PLATINA EXPERIMENTAL port usuario	950
PLACA EXPANSION PORT USUARIO (3 salidas)	3.900
EXPANSION PARA 4 CARTUCHOS	10.900
VENTILADOR PARA DISK DRIVE	3.900
CABLE 40/80 COLUMNAS PARA C-128	2.850
CASSETTE COMPATIBLE C-64 y VIC-20	4.900

SOFTWARE DE GESTION Y UTILIDADES PARA AMIGA, C-64 Y 128. CONSULTENOS

INTEGRADOS Y CONECTORES

CIA 6526	3.100
CPU 6510	3.100
SID 6580	5.000
PAL 6569	6.000
PLA 906114	3.100
8502	4.500
8721	4.500
8701	3.100
CONECTOR PORT USUARIO	750
CONECTOR PORT EXPANSION para placa	1.250
EPROM 27128	990

PEDIDOS POR



(93) 224 34 22

PROGRAMADOR DE EPROMS	14.900
IC TESTER	16.900
TARJETA DE EPROMS 4 x 8 K	3.250
BORRADOR DE EPROMS	15.900
OSCILOSCOPIO para C-64 y 128	24.900
DISK NOTCHER (Taladro doble cara diskettes)	1.950
RATON C-64 (soft en diskette o cassette)	9.900
CINTA IMPRESORA MPS-801 y GP/500	950
CINTA IMPRESORA STAR SG-10 y OKI 80	250

CONDICIONES ESPECIALES PARA DISTRIBUIDORES

SERVICIO DE REPARACIONES



REPARACION DE C-64, C-128, C-16 Y VIC-20
DISK DRIVE 1541, 1570 Y 1571 REPARACION Y AJUSTE.
REPARACION Y AJUSTE DATASSETTES.

SOLICITE NUESTRO CATALOGO
ENVIOS CONTRA REEMBOLSO A TODA ESPAÑA SIN GASTOS.
ENVIOS INFERIORES A 2.000 PTAS., AÑADIR 300 PARA GASTOS DE ENVIO.
SUDAMERICA AÑADIR 700 PTAS.
ACEPTAMOS TARJETAS DE CREDITO.





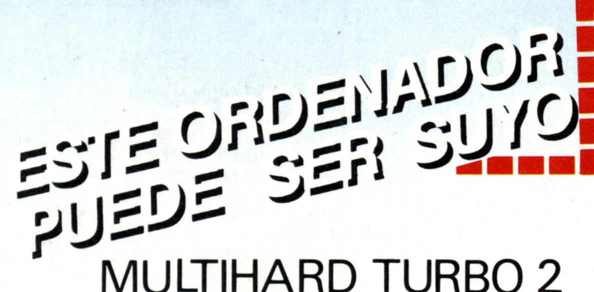
FLORIDABLANCA, 54 Ent. 3.º A
08015 BARCELONA
Tel. 224 34 22

NOVEDAD

16 TOMOS

CON MAS DE
40.000 TERMINOS
Y 2.000 PAGINAS
CON ILUSTRACIONES
A TODO COLOR

VENTA EN QUIOSCOS



MULTIHARD TURBO 2

Recorte y envíe este cupón de solicitud de información

antes del 19 de abril

a Ediciones INGELEK, apdo. de Correos 61294. 28080 Madrid

Nombre ____________________
Dirección ____________________
C. Postal ________ Población ____________
Provincia ____________________







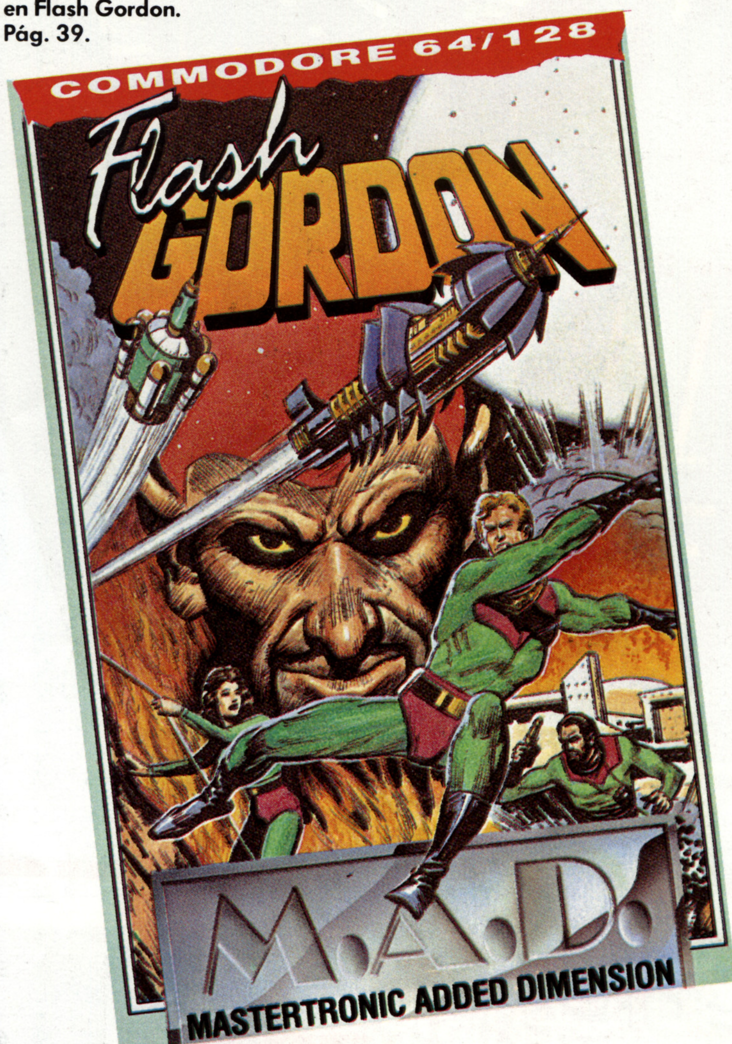
Director:
ANTONIO M. FERRER ABELLO
Redactor Jefe:
FERNANDO LOPEZ MARTINEZ
Redacción:
ANTONIO CARVAJAL
JOSE LUIS DE DIEGO
JUAN M. LOPEZ MARTINEZ
PABLO GARCIA MOLINA
IGNACIO BARCO LUENGO
ALFREDO SINDIN VALERO
FERNANDO ACERO MARTIN
Colaboradores:
JOSE LUIS M. VAZQUEZ DE PARGA
ANTONIO MANZANERA
Secretaria de Redacción:
PILAR MANZANERA AMARO
Maquetación:
CARLOS GONZALEZ AMEZUA
CARLOS TALLANTE
Ilustraciones:
ANTONIO PERERA
RAMON POLO
Fotografía:
EQUIPO GALATA
Directora Publicidad:
CARMINA FERRER
Publicidad Madrid:
BEGOÑA LLORENTE
Tel.: (91) 457 69 23
Publicidad Barcelona:
ISIDRO IGLESIAS
Avda. Corts Catalanes, 1010
Tel.: (93) 307 11 13
Director de Producción:
VICENTE ROBLES
Directora de Administración:
MARIA ANTONIA BUITRAGO
Suscripciones:
MARIA GONZALEZ AMEZUA
Redacción, administración, publicidad y suscripciones:
Plza. República Ecuador, 2.
28016 MADRID. Tel.: 457 94 24 (3 líneas)
Télex 49371 ELOC E
Dirección para correspondencia:
Aptdo. de Correos 61.294
28080 MADRID
TU MICRO COMMODORE es una publicación mensual de Ediciones INGELEK.

Fotomecánica:
RODACOLOR, S. A.
Impresión:
GRAFICAS REUNIDAS, S. A.
Distribución:
COEDIS, S. A.
Valencia, 245. Tel.: 215 70 97
08007 BARCELONA
Almacén: Nacional II. Km. 609,4
MOLINS DE REI (Barcelona)
Delegación en Madrid: Serrano, 165.
Tel.: 411 11 48
Almacén: Laforja, 19-21, esq. Hierro
Pol. Industrial Loeches
TORREJON DE ARDOZ (Madrid)

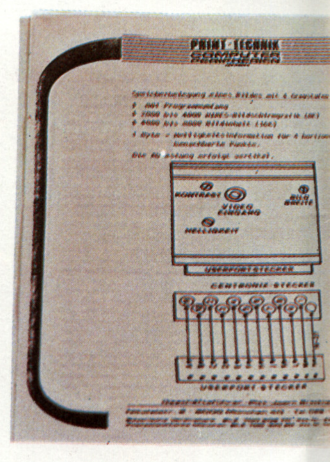
Precios para España. Ejemplar: 350 ptas. (IVA incluido). 330 ptas. (Canarias, Ceuta y Melilla).
(La suscripción anual incluye 11 números).
Distribución Cono Sur:
CADE, S.R.L.
Pasaje Sud América, 1532
Tel.: 21 24 64
Buenos Aires 1.290. Argentina.

Depósito Legal: M. 40920-1985.
Impreso en España
abril 1987

SOFTMODORE
Quedan 24 horas para la destrucción mundial; nuestras esperanzas están depositadas en Flash Gordon.
Pág. 39.

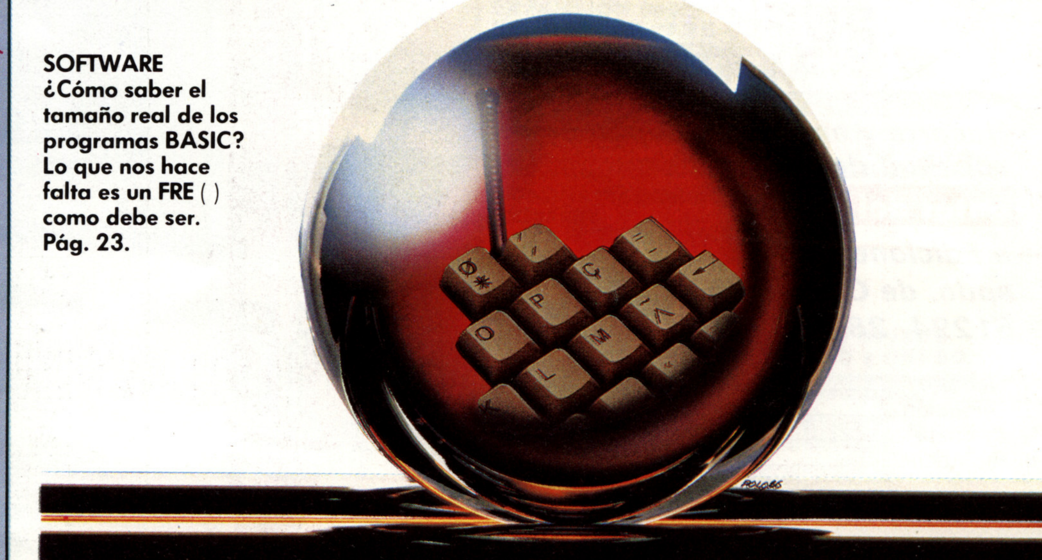






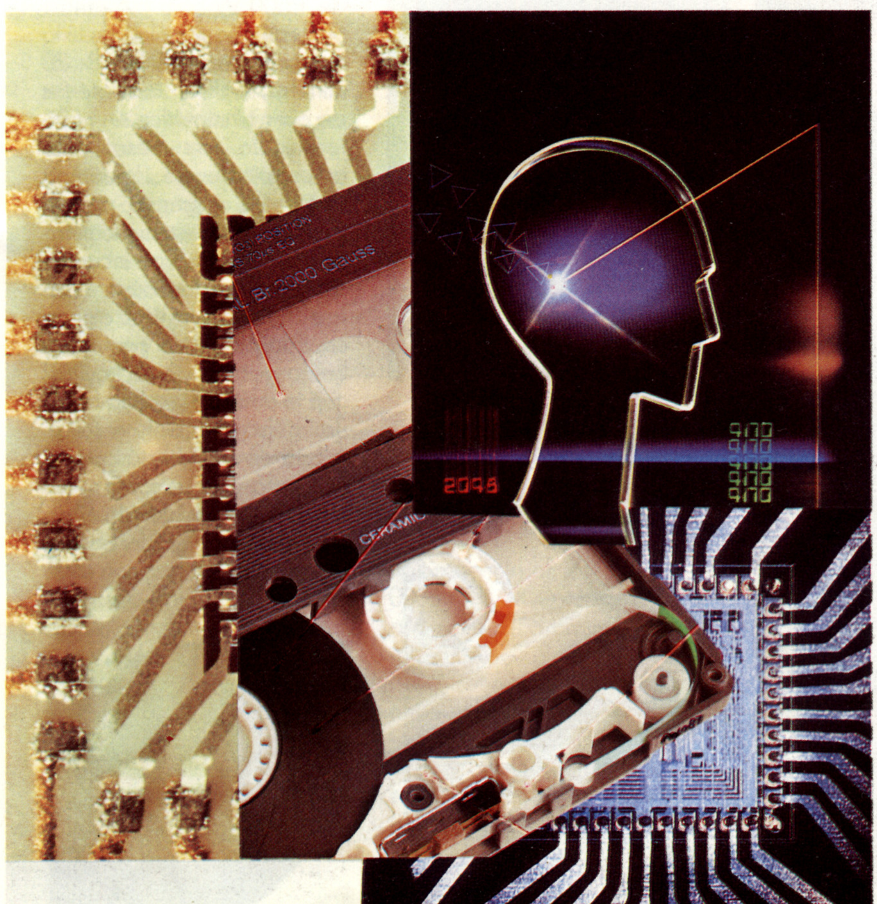
EQUIPOS
Si disponemos de un C-64, ya podemos tener nuestro digitalizador de imágenes con Digi-View 64.
Pág. 26.

SOFTWARE
¿Cómo saber el tamaño real de los programas BASIC? Lo que nos hace falta es un FRE ( ) como debe ser.
Pág. 23.





ALTO NIVEL
Este mes explicaremos cómo se guarda la información en el datassette. Pág. 44.



SUMARIO

6 **NOTICIAS.**

8 **A FONDO.**
Aliens.

16 **EL PROFESIONAL.**
Laser Compiler.

19 **TECLA A TECLA.**
Alkerke.

23 **SOFTWARE.**
Un FRE() como debe ser.

26 **EQUIPOS.**
Digi-View 64.

30 **CONCURSO DE PROGRAMACION.**

31 **CURSO DE BASIC.**
Sprites y otras características gráficas.

39 **SOFTMODORE.**
Jail Break. Microrhythm. Mad Nurse. 180. Flash Gordon.

44 **ALTO NIVEL.**
Datassette.

50 **HOT LINE.**

52 **TECLA A TECLA.**
El Bombero.

56 **FUERA ERRORES.**

58 **TABLON.**

59 **CODIGO MAQUINA.**
Kernal.

62 **LOS 7 MAGNIFICOS.**

66 **CHISPAS.**

Cumplimos con este número 12 un año de nuestra cita mensual en el quiosco, en esta nueva época nacida de vuestras opiniones y sugerencias sobre el contenido, presentación y talante de la revista. Aprovechamos pues la ocasión para felicitaros, y agradeceros vuestra colaboración en la confección de TU MICRO COMMODORE, pues ya hemos comenzado a recibir contestaciones al cuestionario publicado el mes pasado. Animaos y seguid enviando vuestras respuestas; tenéis aún de plazo hasta el 15 de abril para optar a ese fantástico sorteo de una cadena Hi-Fi y diez Koala Pads.

noticias

Premiados en el concurso Serma-Galerías Preciados

A continuación publicamos la lista de ganadores en el sorteo Serma-Galerias Preciados, llevado a cabo durante la campaña de Navidad. Los premios consisten en cinco motos Suzuki-Puch que serán entregadas próximamente a:

Antonio Magana, de Sevilla.

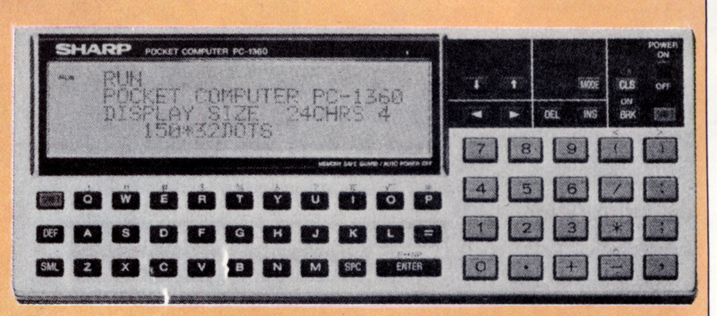
Manuel Castilla, de Granada.

Juan Carlos Melián Sosa, de Las Palmas.

Pedro Milla Márquez, de Córdoba.

Juan Antonio López Alcaraz, de Murcia.

Ordenador de bolsillo Sharp con 64KB



La empresa Sharp presenta un nuevo miembro de su amplia gama, el PC-1360, el cual incorpora todas las funciones que se precisan para aplicaciones profesionales y personales.

Este nuevo ordenador es programable en BASIC, e incluye funciones matemáticas y funciones de edición ampliadas (renum, delete). Dispone de los slots para tarjetas de memoria RAM intercambiables, que permiten disponer de hasta 64Kb de memoria RAM, para crear sofisticados y potentes programas.

Incorpora interfaces para recogida y transmisión de información a través del RS-232 que, en sus reducidas dimensiones, combina las funciones de una calculadora con las posibilidades de un ordenador programable.

La pantalla del PC-1360 nos ofrece 4 líneas de 24 carecteres. Trabajando con gráficos podemos controlar desde el BASIC cada uno de los 150 x 32 puntos.

El Sharp PC-1360 se importa por Mecanización de Oficinas S.A. y se distribuye a través de distribuidores autorizados.

Enfa Ibérica cambia de domicilio

Enfa Ibérica, empresa especializada en periféricos y microaccesorios imprescindibles para nuestro ordenador, cambia de domicilio a la calle Senda Galiana s/n. Edificio Enfa. 28080 Coslada (Madrid), y su nuevo teléfono 672 72 11.

Novedad de ámbar electrónica: Olimpic 2000

El sistema de alimentación ininterrumpida OLIMPIC 2000 es muy útil para evitar la pérdida de información que estemos tratando en nuestro ordenador en el caso del corte eléctrico. Es totalmente silencioso, con un tamaño y peso reducido. Posee además dos tipos de salida, de onda sinusoidal y permanente sin cortes, con una potencia nominal con coseno phi 0,7 200 VA y una potencia nominal de 1.500 W.

Su autonomía al 75% de la carga es de un mínimo de 20 minutos, utilizando baterías de plomo hermético sin mantenimiento. Se desconecta automáticamente si la batería está baja, avisando previamente con un sistema luminoso. Así mismo, tiene incorporado un avisador acústico para el fallo de suministro eléctrico. Está protegido contra sobrecargas y cortacircuitos, con la entrada y salida aisladas galvánicamente entre sí.

Esta novedad está comercializada por Ambar Electrónica, sito en la Avenida Marqués de Motroig, 259 de Badalona (Barcelona), con el teléfono (93) 211 69 68.



La base de datos Fundesco sobre tecnologías de la información: útil y gratuita

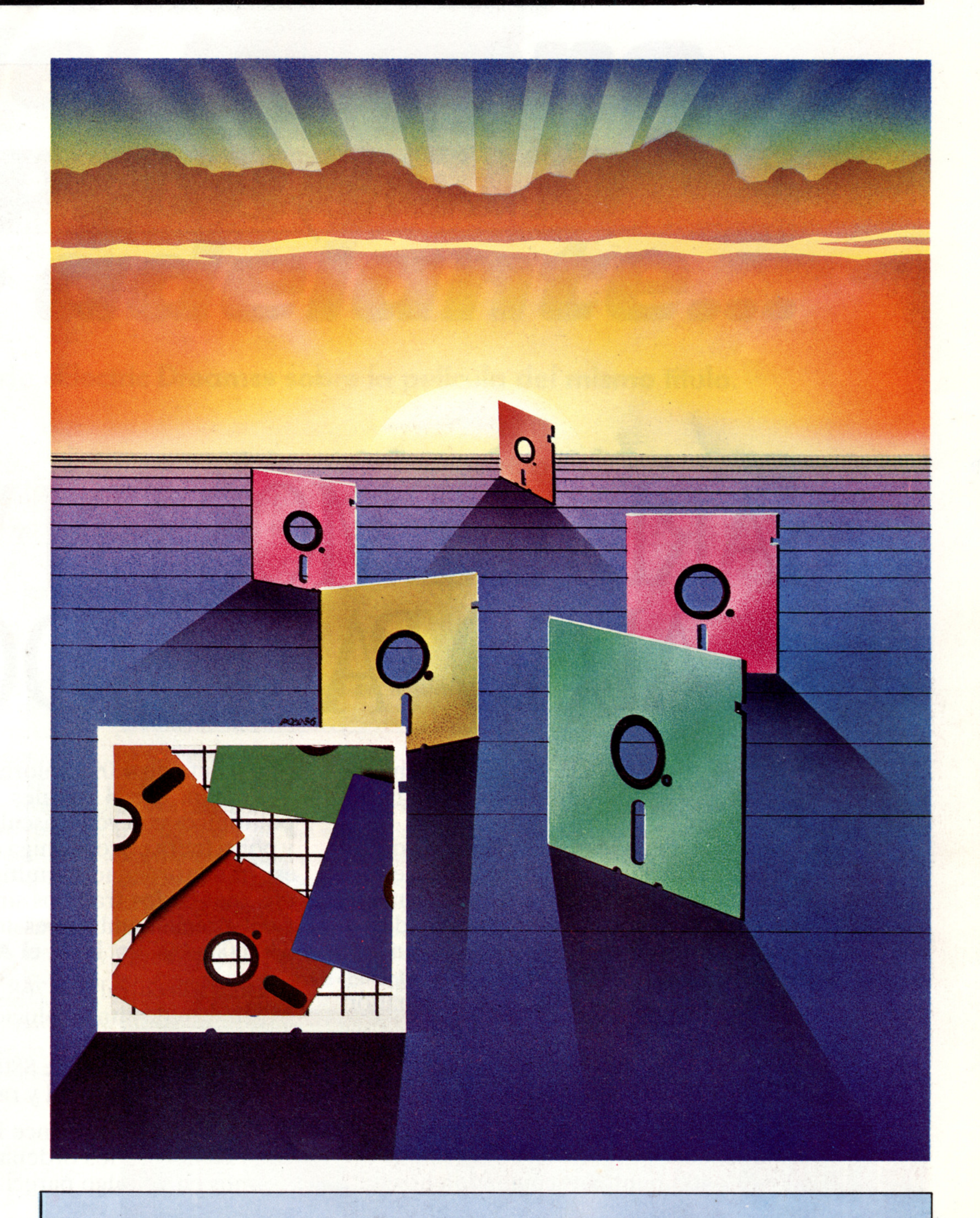
La base de datos Fundesco, creada por la Fundación para el Desarrollo de la Función Social de las Comunicaciones, alberga lo que se publica en castellano en las revistas del sector de Telecomunicaciones, Electrónica e Informática, además de reseñas de proyectos de investigación, tésis, informes, libros, etc., de gran utilidad para aclarar un concepto o enterarse de las tendencias más modernas.

Conectarse en línea a la base de datos Fundesco es muy fácil: basta disponer de un ordenador personal con un programa de comunicaciones, un modem y por supuesto, un teléfono. Si a esto añadimos la gratuidad, por lo menos hasta el 10 de junio de 1987, es evidente lo ventajoso de su oferta.

Además, para aquéllos que no dispongan del mínimo equipamiento necesario para conectarse por teléfono, existen el servicio fuera de línea, que se solicita por carta o por teléfono, y el de «difusión selectiva de la información», que es una especie de «abono» para recibir periódicamente las últimas informaciones introducidas en la base de datos sobre el tema o temas de interés para el abonado.

Para solicitar más información o, simplemente, la inclusión en el registro de usuarios y recibir la correspondiente contraseña y los documentos de consulta, dirigirse a:

FUNDESCO
Serrano, 187
28002 Madrid
Tel.: (91) 4505800



Enhorabuena a los ganadores

Este mes los agraciados ganadores de una suscripción por seis meses a nuestra revista TU MICRO COMMODORE, por la ayuda prestada en la confección de la sección LOS 7 MAGNIFICOS, han sido los siguientes lectores:

José Pascual, de Sabadell (Barcelona).

M.ª Sol Marota, de Alcalá de Henares (Madrid).

Mariano Arrese, de Madrid.

Ivan Nieto, de Bilbao.

Nuria Draper, de Lloret de Mar (Gerona).

Elia Dominguez, de La Coruña.

José Ferrón González, de Mérida (Badajoz).

José Ramón Quesada, de Madrid.

José García Caro, de Valls (Tarragona).

José Ma. Gómez, de Aguilas (Murcia).



TARDAREMOS EN VER ALGO PARECIDO.

En informática ya es difícil sorprender. Pues bien, Commodore lo ha conseguido con Amiga.

Más que una nueva generación de ordenadores, el Amiga de Commodore representa un nuevo concepto. Un ordenador que ofrece una serie de posibilidades y abre unos caminos que hasta hoy eran impensables.

Entre las novedades de este ordenador profesional destacan: la posibilidad de trabajar con un procesador de 32 bits y de 3 coprocesadores específicos, actuando los 4 al tiempo.

Además, Amiga dispone de pantallas y ventanas configurables con una resolución de 640 x 400 puntos escogiendo entre una paleta de 4.096 colores.

Trabaja con 4 canales independientes polifónicos con voz masculina y femenina, y por si fuera poco, Amiga de Commodore es el único ordenador multitarea que puede efectuar varios trabajos simultáneamente.

El precio también es importante. Por 330.000 ptas. ex. IVA, el Amiga incluye:

- Unidad central con 768 K RAM.
- Monitor de alta resolución en color y sonido.
- Unidad de discos de 880 K.
- Teclado profesional y ratón.

Por mucho que avance la investigación en el campo de los ordenadores, tardaremos en ver algo parecido.



Imagina siempre lo mejor.

Microelectrónica y Control, S.A. *Unico representante de Commodore en España.*
c/ Valencia, 49-53 Tel. (93) 325 57 02 08015 Barcelona - c/ Ardemans, 8 Tel. (91) 246 37 36 28028 Madrid

Aliens: del noveno pasajero en adelante...

El juego de «Electric Dreams» sobre la película del mismo título.

La nave espacial Nostromo, en misión científica por el Universo, detecta una alarmante emisión de naturaleza desconocida procedente de un planeta cercano. La tripulación decide investigar y descubre unas extrañas formaciones geológicas que resultan ser embriones de una voraz y temible raza de seres alienígenas. Uno de estos extraterrestres consigue introducirse en la nave, sembrando el terror y la muerte entre los tripulantes. Se trata de... ALIEN.

Un miembro de la expedición, la valerosa Ripley, logra escapar de la nave antes de que el sistema de autodestrucción entre en funcionamiento. EL ALIEN, finalmente, perece abrasado por las toberas de la cápsula de escape y la nave se aleja mientras un cartel que dice «The End» nos indica que es hora de abandonar la sala.

Aunque en un principio así lo creíamos, la historia no terminó con el citado letrero. Unos años después, la cápsula en la que viajaba Ripley llegó a la Tierra, y ésta fue interrogada. La historia del marcianito feucho y hambriento no fue demasiado del agrado de sus superiores, que decidieron no creer ni una palabra y destinarla a otro tipo de actividades.

Al cabo de un tiempo se recibieron noticias del exterior: una base terrestre en un lejano planeta estaba infestada de extrañas criaturas espaciales, con unas costumbres culinarias en extremo desagradables para la especie humana. De modo que Ripley marcha al planeta para verse de nuevo cara a cara con los aliens (todo esto es lógico, o si no, ¿cómo podrían haber realizado la continuación?).

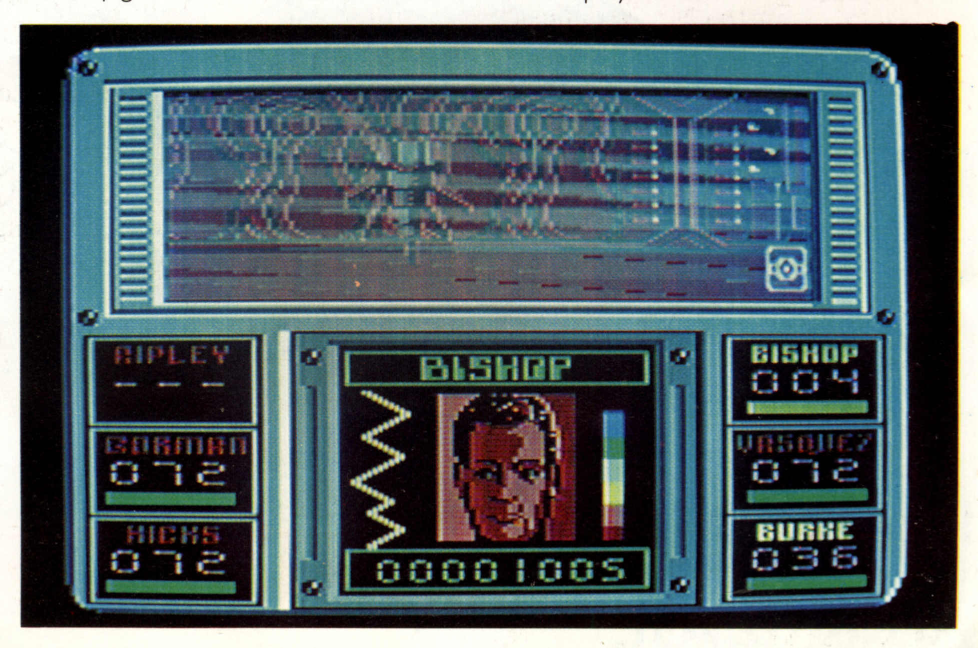
Traslademónos ahora de la pantalla de cine al pequeño monitor de nuestro Commodore, para saber cómo acabará esta aventura (tal vez éste sea el fin definitivo).

«Bueno, ya estamos en la base. ¿Y ahora qué?

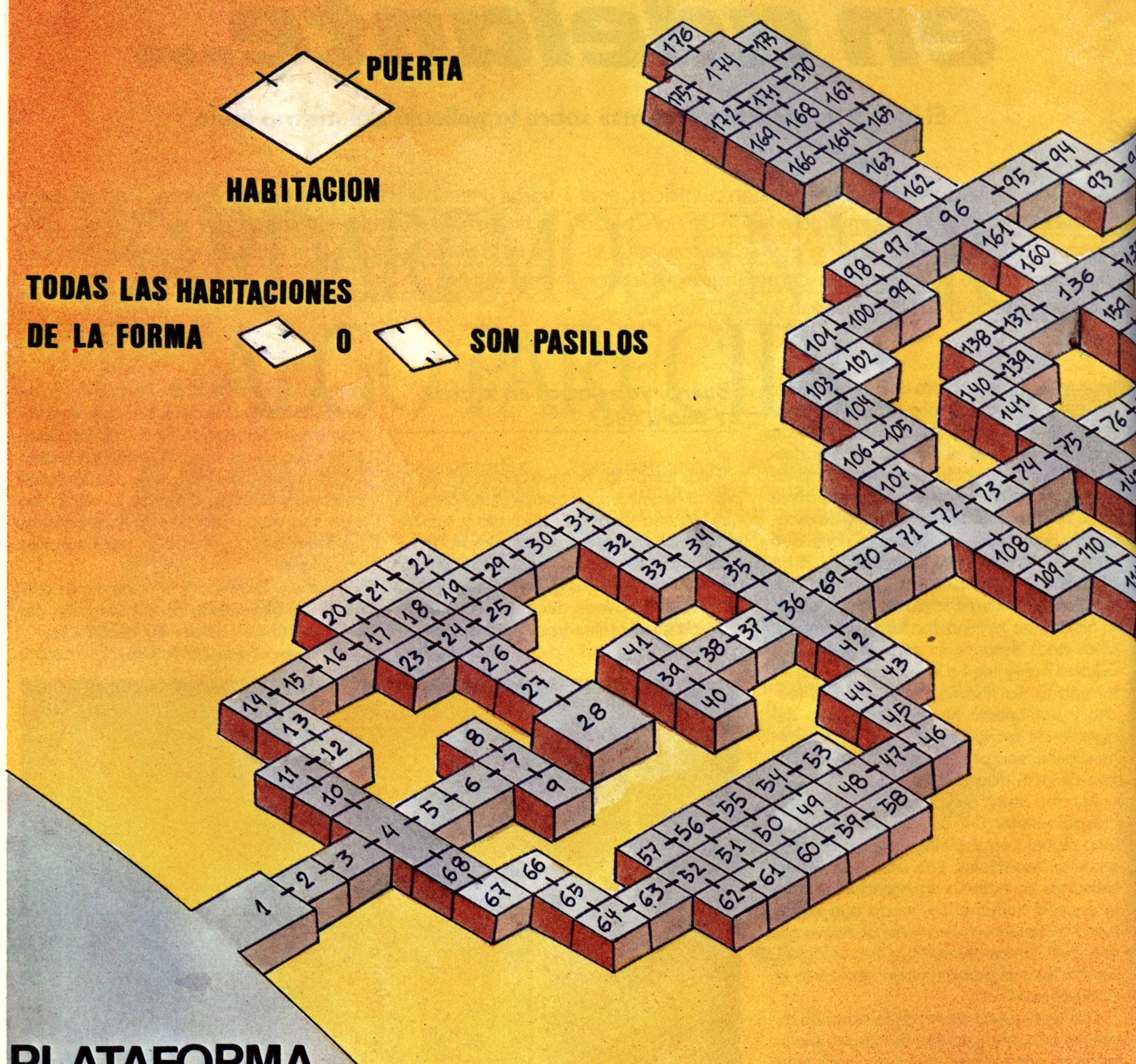
Estas fueron las primeras palabras de Ripley al penetrar en el complejo metálico, lo que da una idea del estudiado y estricto plan de ataque que ya tenía preparado. La verdad es que si un solo alien había causado todos los estragos que ya conocemos... bueno, no se puede decir que al reunirse unos cuantos centenares de ellos se dedicasen a jugar al mus o a comentar el último consejo de alienistros. El estado de la base, junto con el hecho de saber que posiblemente en la habitación de al lado los aliens, enterados de su llegada, estén ya redactando el menú del día, no se puede definir como una situación divertida. La misión iba a ser muy dura y peligrosa, y sinceramente, no sabemos a qué esperas para echarles una mano.

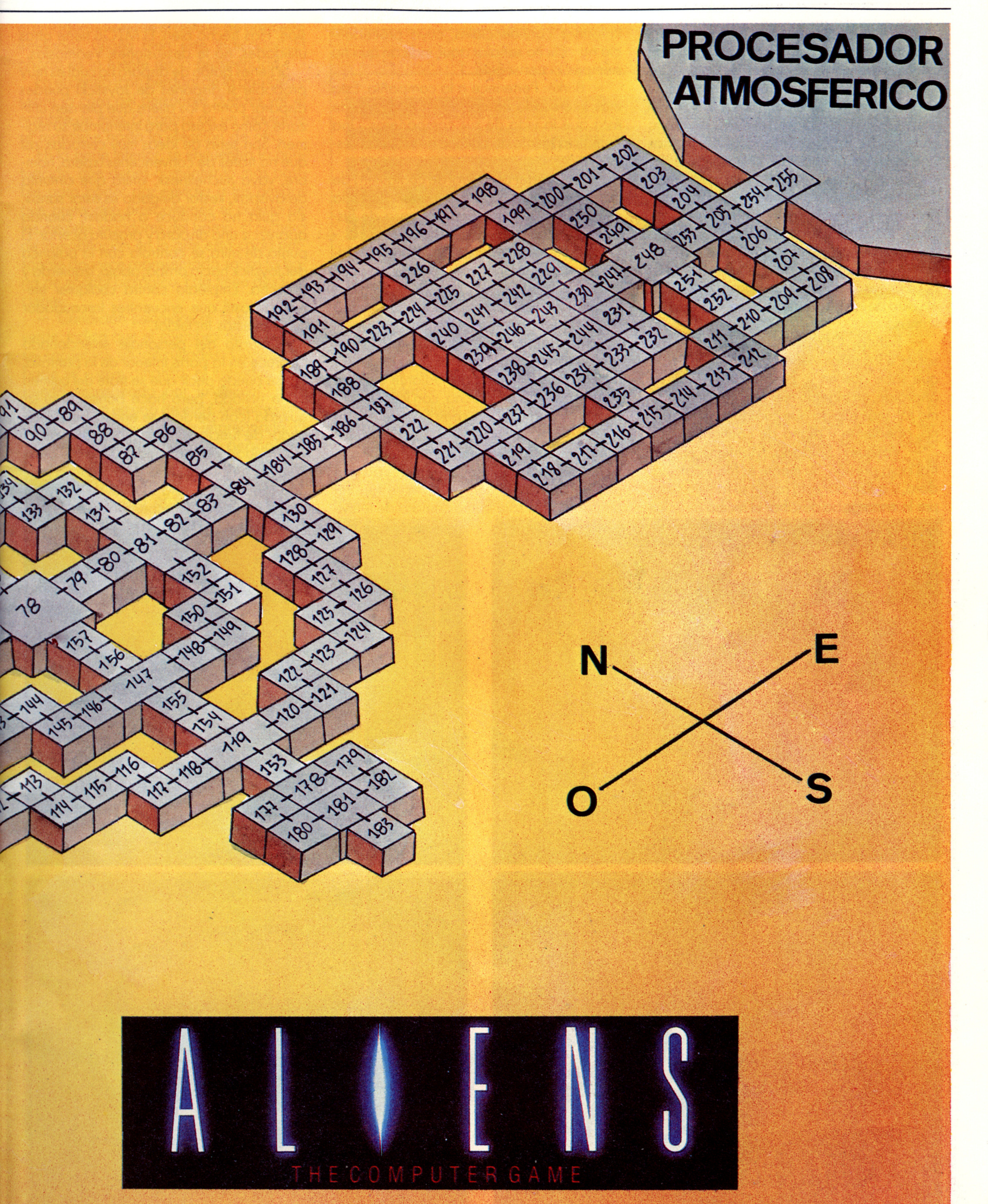
Desde tu ordenador terrestre tienes control sobre 6 miembros de la tripulación, cada uno con sus defectos y cualidades:

— Ripley: Ya la conocemos. Es la única



HABITACION 1 PLATAFORMA MOVIL TACTICO-OPERACIONAL
HABITACION 28 ARMERIA
HABITACION 78 SALA DE CONTROL
HABITACION 174 GENERADOR
HABITACION 177-182 SECCION DE INVESTIGACION MEDICA
HABITACION 248 SALA DE LA REINA




PROCESADOR
ATMOSFERICO
N
E
O
S
ALIENS
THE COMPUTER GAME

superviviente del Nostromo. Está muy afectada por lo que ocurrió y teme a los aliens más que a su propia suegra, aunque lucha contra ellos como si de ésta se tratara.

— Gorman: Teniente y responsable del grupo. Esta es su primera misión y por tanto se le puede excusar el temblor de piernas que le asaltó al entrar en la base y ese desagradable olor de connotaciones metabólicas que todos pudieron apreciar y cuya procedencia nadie trató de averiguar por compañerismo.

— Hicks: Marine colonial. Posee rápidos reflejos y una mente despierta, no se deja acobardar fácilmente, es veloz y suele salir él solito de las peores situaciones. Una joya, vamos...

— Bishop: Oficial. Nunca le preocupó saber si los niños vienen o no de París, porque sabe que en su caso no fue así. Es un androide, un ingenio mecánico, y actúa como tal: más rápido que cualquier ser humano, obediente y digno de confianza.

— Vasquez: Soldado. Mantiene el control bajo presión. Su mente es fría y calculadora, y quizá por esta razón su familia siempre le encargaba descifrar el recibo de la luz (Informes confidenciales, cuya procedencia no es lícito señalar aquí, indican que no siempre lo conseguía...).

— Burke: Pertenece a la compañía. Esta más interesado en conservar el dinero de ésta que la vida de sus compañeros. Una auténtica mente de político en este caos espacial.

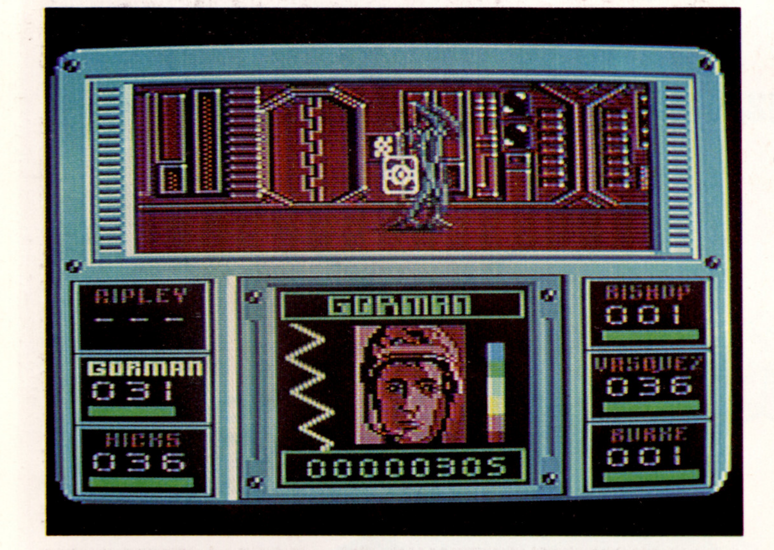
Para acceder al control de cualquiera de estos personajes basta con pulsar la tecla de su inicial, exceptuando el caso de Bishop, cuya letra es la «I».

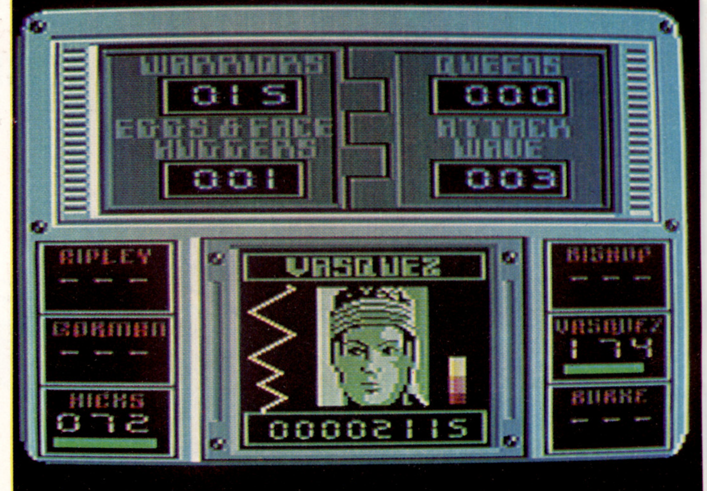
Cada uno de los miembros de la tripulación está equipado con un MTOB, cuyo bonito nombre significa Mobile Tactical Operations Bay, más o menos. Gracias a este sistema podemos tener una vista parcial de la habitación en la que el personaje se encuentra. Esta panorámica puede ser desplazada a voluntad mediante el Joystick a izquierda o derecha, con lo que conseguimos una imagen completa del sector en cuestión, así como ver acercarse a los aliens antes de que sea demasiado tarde.

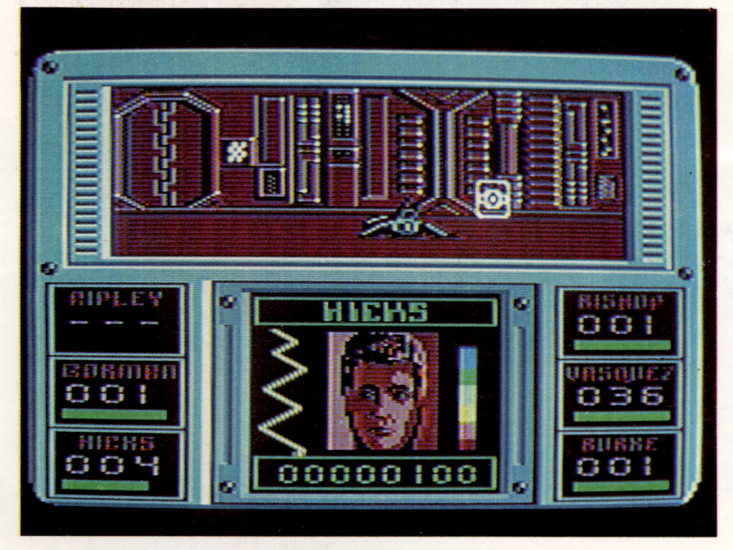
Te estarás preguntando cómo es posible que un aparatejo con un nombre tan impactante sirva sólo como cámara de televisión. Has hecho bien. El MTOB tiene otras utilidades, no te desilusiones. Veámoslas.

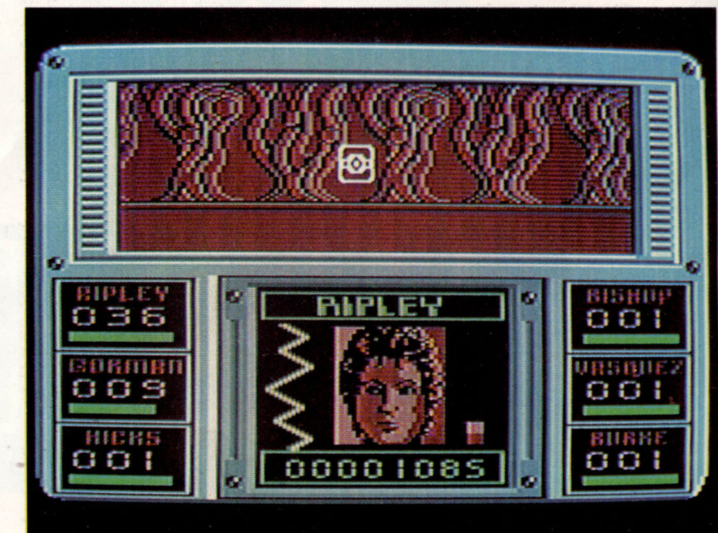
Si tienes botón de disparo (lo cual es muy probable si dispones de Joystick) puedes disparar contra los aliens que piensen utilizarte como excusa para interrumpir su régimen. Son seres de ideas muy cerradas, así que no trates de convencerles («pero, hombre, vaya michelines», «mira que cinturita de avispa», «estar delgado es estar sano», «la gula no conduce a nada»), cuando aparezca uno dispara y procura acertar. El punto de mira se controla con la palanca del Joystick hacia arriba, abajo, izquierda y derecha. Empléalo contra cualquier elemento hostil que encuentres (recaudador de impuestos incluido).

El retrato del personaje seleccionado aparece en la parte inferior de la pantalla. No se puede decir que se trate del culmen de la utilidad, pero hace bonito. A su izquierda (con perdón) se encuentra el indicador de funciones biológicas, que señala la actividad del personaje. Si éste muere, deja de funcionar. A la derecha (con perdón) del retrato, está el nivel de munición (suponemos que no esperarías que los pro-









gramadores tuvieran el detalle de ofrecerla ilimitada). ¡Vigila bien este indicador! El ahorro es la fuente de la prosperidad.

En el centro (con perdón) de la pantalla y sobre la cara del personaje puedes ver el nombre de éste. El control de un tripulante se mantiene hasta la elección de otro.

En la parte inferior, se encuentra la puntuación, que debes aumentar lo más posible para poder dar envidia a los amigos que pretenden arrebatarnos el puesto de los mejores jugadores del barrio. El número de la habitación en la que está cada personaje aparece representado sobre la barra de control biológico. Esta tiene dos funciones:

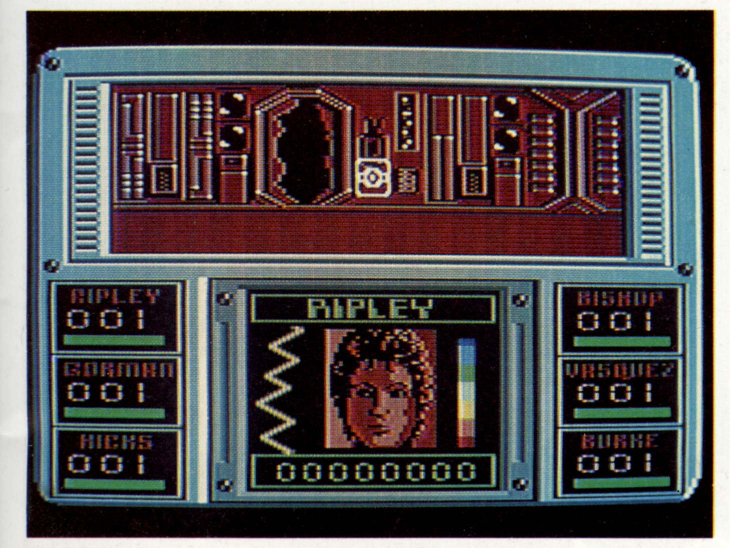
— Condición psicológica: Si el color es verde, indica que el miembro de la misión se encuentra sin dificultades. Cuando la barra se vuelve amarilla, el individuo está siendo impregnado por los aliens, para transformarlo en uno de ellos. Mientras uno vigila, otro se encarga de impregnarlo. Es posible salvarle enviando otro personaje a la habitación y matando al vigilante. Si la barra es de color rojo, ya no hay salvación posible. Reza una oración por el eterno descanso de su alma y sigue tu camino (bucólico ¿eh?).

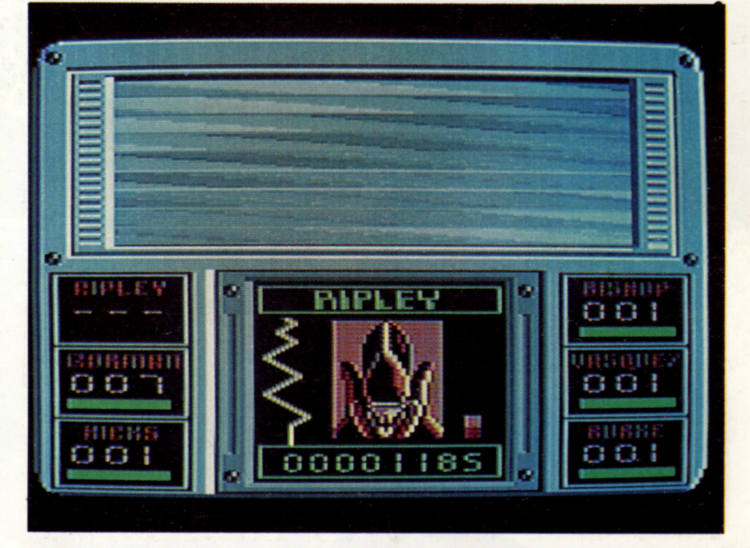
— Nivel de fuerza: Cuanto más se mueva un tripulante más cansado estará, lo que se nota en la disminución de longitud de la barra. Si ésta parpadea, el nivel de cansancio es peligrosamente alto, y si desaparece es necesario seleccionar otro personaje, ya que el escogido está completamente exhausto y no responderá a tus órdenes.

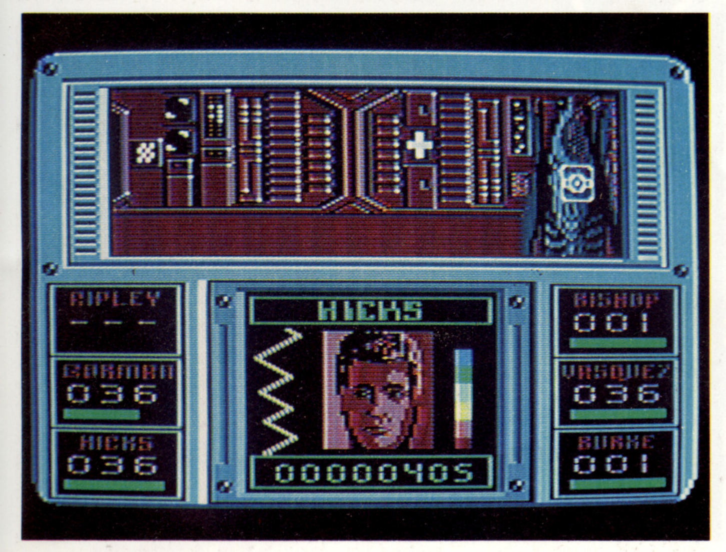
Si de repente empieza a sonar una alarma no pienses que tu ordenador está siendo atracado, lo que ocurre es que el detector de proximidad señala la presencia de un alien en la misma habitación en la que el nombre del tripulante cambie de color. Asume el control de éste lo más deprisa posible y defiéndete con uñas y dientes contra el ataque (un consejo: da mejor resultado hacerlo con el canon láser).

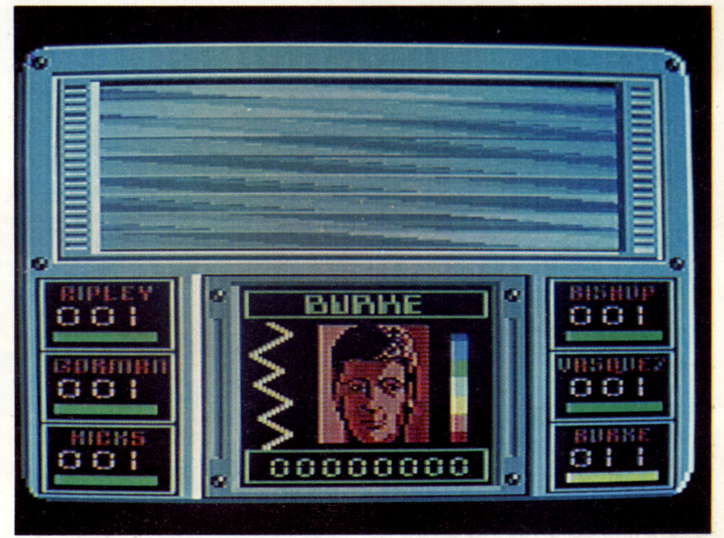
FICHA TECNICA

NOMBRE	ALIENS
PRECIO	880 PTAS.
SOPORTE	CINTA
TIPO	ACCION Y ESTRATEGIA
MODELO	C-64, C-128
OBSERVACIONES	NINGUNA









6 personajes y 2 objetivos: vivir y mantener el control de la base

Esta vez lo han hecho. ¡Por fin! Junto con el juego viene incluido un mapa completo de la base con las habitaciones más importantes numeradas. Las demás están en blanco, pero por tratarse de ti, que nos sigues número a número, publicamos un plano con los números de todas las habitaciones. De nada. Si te sientes lo suficientemente agradecido por darte el trabajo resuelto puedes enviarnos un cheque de seis cifras por lo menos. Será bien recibido.

Una vez tengas tu mapa en la mano puedes moverte de dos formas: situando el punto de mira sobre una puerta y pulsando la barra espaciadora o tecleando el número de habitaciones (de 1 a 9) y la dirección (N, S, E, W) en que deseemos que se desplace. Al tomar el control de otro personaje, el programado se moverá según lo indicado. Las puertas pueden ser selladas disparando contra el dispositivo que se encuentra a su lado. Para pasar por una puerta bloqueada es necesario derribarla antes. Los aliens entran por todos los lados, esté la puerta sellada o no, por lo que esta medida no te protegerá de su voracidad.

Los excrementos bio-mecánicos cubren las paredes de las habitaciones por donde han pasado los aliens. Si no las destruyes crecen y tapan las puertas, no dejándote pasar. A pesar de la desolación que reina en la base, Newt, una pequeña niña ha conseguido sobrevivir, escondiéndose en los tubos de ventilación. Puede aparecer en cualquier momento, y si la encuentras recibirás una puntuación extra.

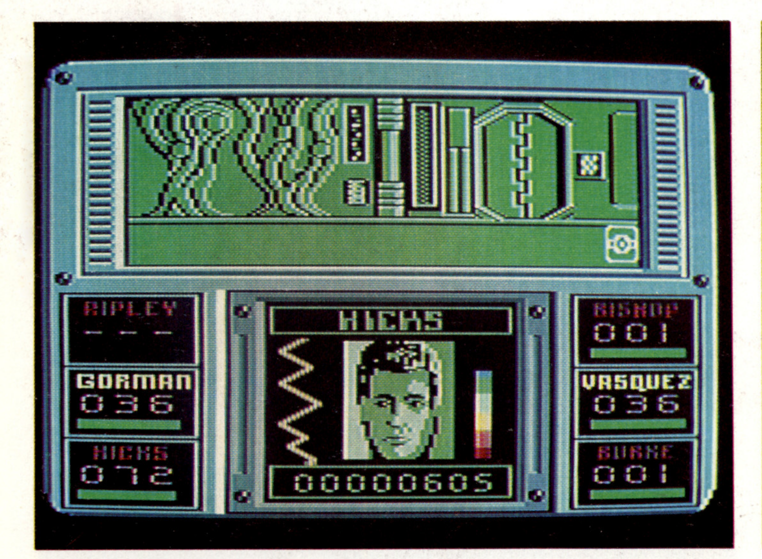
En todo momento, pulsando la letra «P», puedes producir una pausa y consultar una tabla resumen de los aliens destruidos y la oleada de ataque en curso.

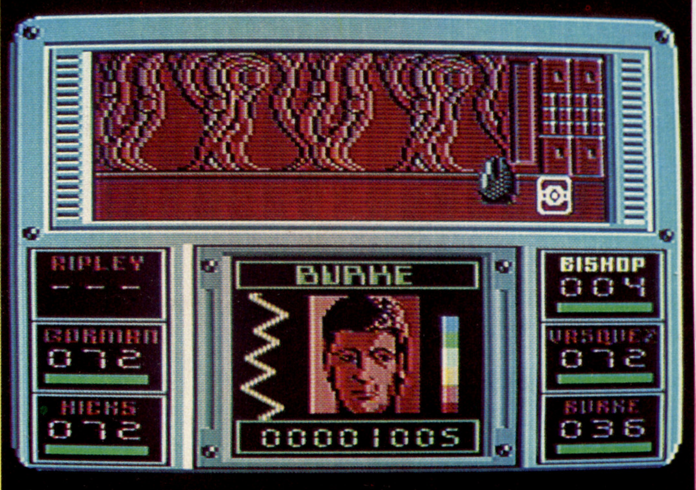
Algunas habitaciones son claves para el desarrollo del juego. El centro de control (78) y el generador (174) son muy importantes. Sitúa algunos tripulantes aquí, pues si pierdes el control de estas habitaciones las luces se apagarán. En la número 248 está la reina, poniendo huevos continuamente. Lleva hasta aquí la mayor cantidad posible de gente y ¡que la fuerza te acompañe!

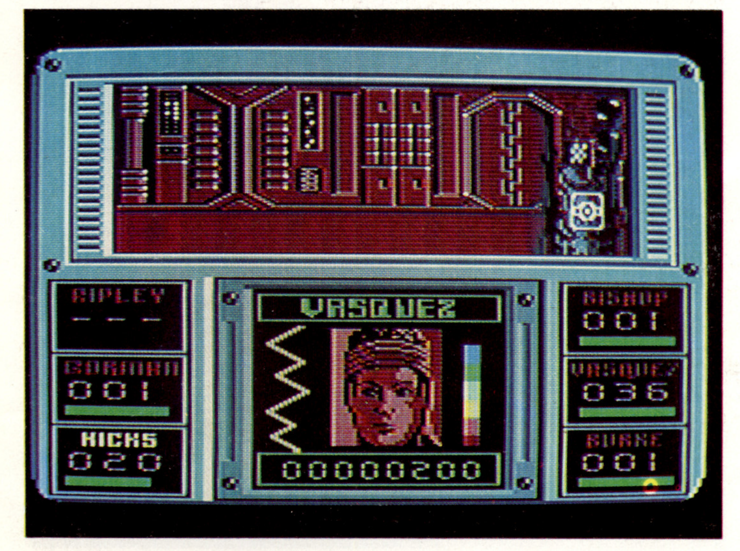
¿Quién dijo que segundas partes nunca fueron buenas?

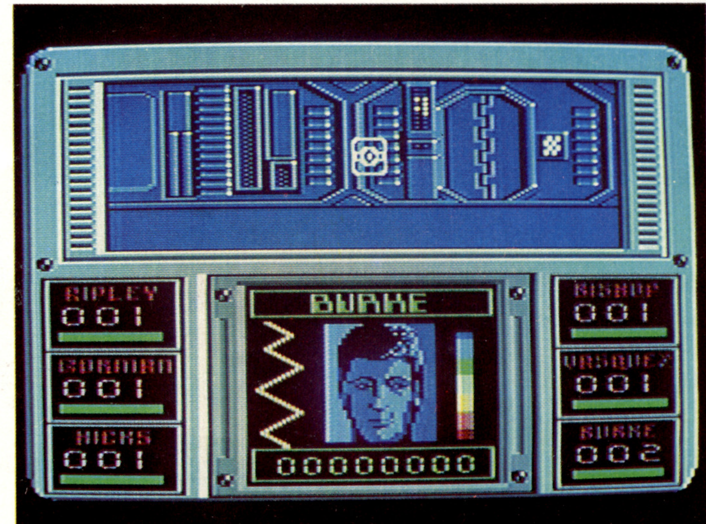
Aliens es un buen juego, ni más ni menos. La calidad gráfica está muy bien conseguida, así como el desarrollo y forma de manejo. En un principio es bastante entretenido, pero con el paso del tiempo llega a aburrir un poco. No se puede decir que se trate de un juego indispensable en cualquier ludoteca, pero está de más tenerlo a mano para jugar de vez en cuando.

Queremos aquí hacer un llamamiento a los programadores y distribuidores de juegos: ¡por favor, no nos pongan las cosas difíciles! Agradecemos enormemente el gesto de incluir el mapa con el programa (de otro modo sería prácticamente imposible jugar), pero podrían haberlo rellenado ¡que lo tenemos que hacer todo nosotros! Sobre todo, tratándose de juegos difíciles como éste, apoyamos la idea de los planos incluidos en el precio, y esperamos que pronto se convierta en una deseable costumbre. Gracias.









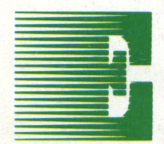


La Informática tratada a fondo en 40 monografías
¡Y LLEGO EL BASIC! (I)
DIMENSION MSX
¡Y LLEGO EL BASIC! (II)
INTRODUCCION AL PASCAL
PROGRAMANDO COMO ES DEBIDO
SOFTWARE DE BASE
OPERATIVO CP/M
DEL ORDENADOR
DE APLICACIONES
VISICALC
EL ORDENADOR
DE TEXTOS
DE JUEGOS
LOGO
INTEGRADOS
II Y DBASE III
DE DATOS (I)
BANCOS DE DATOS (II)
FORTH
BASIC Y TRATAMIENTO DE IMAGENES
LOS ORDENADORES UNO A UNO
MULTIPLAN
SYMPHONY
INTELIGENCIA ARTIFICIAL Y SISTEMAS EXPERTOS
PROGRAMAR EN C
MUSICA Y SONIDO EN SU ORDENADOR
FORTRAN Y COBOL
BIBLIOTECA BASICA INFORMATICA
MS/DOS 2
COMO ELEGIR UN ORDENADOR 1
INGELEK
EDICIONES INGELEK
B. B. I.

Láser Compiler

Un compilador para algo más que el simple BASIC

Este compilador, desarrollado por OASIS SOFTWARE, es totalmente compatible con el intérprete de LASER BASIC, versión ampliada de este lenguaje, en la que se incluyen 256 nuevas palabras BASIC para los COMMODORE.



El compilador traduce tanto los programas escritos en LASER BASIC, como los confeccionados por la versión de este lenguaje contenida en la ROM del propio ordenador, obteniendo como resultado un programa «objeto» o «absoluto», en código máquina, ejecutable por el ordenador a una velocidad muchas veces superior a la que cabe esperar de la puesta en funcionamiento de programas BASIC en modo intérprete.

Otra característica peculiar de esta aplicación es la de poder compilar programas fuente de cualquier longitud, incluso si éstos se encuentran almacenados en cinta. En cualquier caso, a la hora de compilar un programa es posible optar por una serie de alternativas:

1. Aritmética entera, con objeto de ahorrar en tamaño del programa C/M y velocidad de ejecución, o aritmética habitual de coma flotante.
2. Conexión/desconexión del chequeo de errores de compilación.
3. Compilación de programas fuentes escritos en lenguaje BASIC convencional o LASER BASIC.

El compilador es totalmente compatible con el BASIC intérprete, facilitando el empleo de cadenas y tablas de longitud variable. Además, es posible compilar, a su vez, la mayoría de los programas BASIC escritos en LIGHTNING BASIC, bajo las restricciones especificadas en el manual.

Los programas objeto pueden cargarse y ejecutarse, con total independencia del software del compilador, permitiendo a los autores la libertad de publicar los programas absolutos sin restricciones comerciales.

Otra ventaja adicional, para los usuarios de la unidad de cinta, es la de poder beneficiarse de la rutina de lectura/grabación rápida de que está dotado el compilador. De esta forma, los programas objeto almacenados en cinta trabajan a gran velocidad.

Contenidos del programa

Además del programa compilador, se suministran, en cada disco o cinta, una serie de programas de utilidad:

1. DEMO muestra los resultados que se pueden obtener con el uso combinado del LASER BASIC y el compilador.
2. TURBO es una rutina de utilidad, destinada a los usuarios de cinta, la cual facilita la lectura/grabación rápida, según las indicaciones contenidas en el manual.
3. BOOTGEN se suministra únicamente en la versión en disco. Permite almacenar programas sobre este soporte, los cuales pueden cargarse en la forma:

LOAD "nombre-fichero",8,1

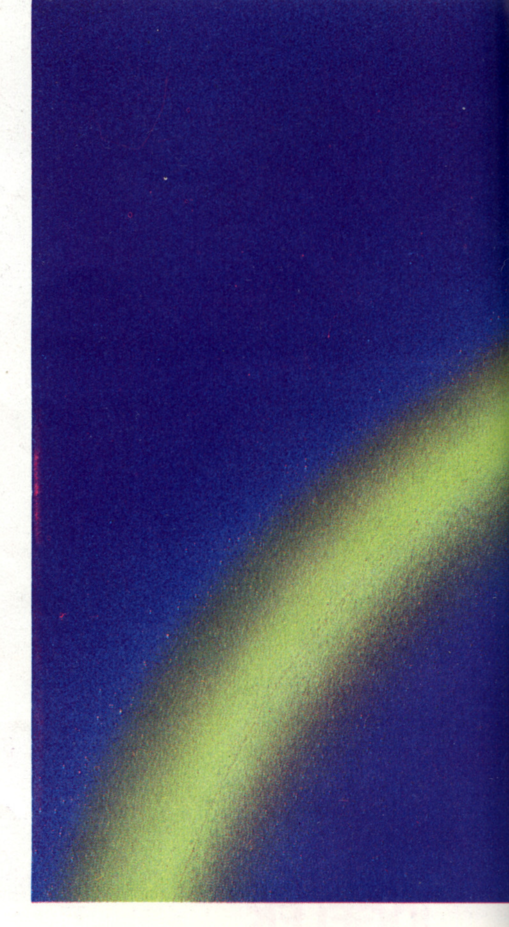
4. LIB1 a LIB6 son librerías en tiempo de compilación, las cuales deben estar presentes para poder compilar cualquier programa, aunque sólamente es necesaria una de ellas cada vez, dependiendo de las opciones de compilación especificadas.

Funcionamiento del compilador

Nada más poner el programa compilador en funcionamiento, éste nos pregunta por el número de periférico con el que vamos a trabajar. Las opciones posibles son: 1-cinta, 8-disco, 9/10/11 unidades de disco adicionales, obteniéndose por defecto, al pulsar **RETURN,** el número de periférico correspondiente a la unidad desde la cual se ha efectuado la carga del programa compilador.

A continuación, se pide el nombre del programa BASIC fuente, el cual es cargado en la memoria. Inmediatamente, aparecen en la pantalla las diferentes opciones a seleccionar, suponiéndose esta vez por defecto, que se trata de programas LASER BASIC con aritmética de coma flotante.

A partir de este momento, es cuando co-



mienza verdaderamente el proceso de compilación, el cual requiere de tres pasadas de las que, en las dos primeras, obtendremos en pantalla información relativa al número de línea en tratamiento. Si no se producen errores, obtendremos el mensaje de compilación satisfactoria, al final de la tercera pasada.

Para completar la compilación es preciso cargar uno de los módulos de biblioteca, dependiendo de las opciones de compilación especificadas. En el caso de la unidad de disco, esta carga se efectúa de forma inmediata; en el de la unidad de cinta, se pregunta primero por la velocidad de carga, por si utilizamos o no, en su momento, la función TURBO. Completada esta operación, el programa objeto queda convertido en ejecutable.

Palabras BASIC no utilizables

El empleo de determinado vocabulario de palabras BASIC, resulta absurdo o incompatible con el funcionamiento del programa compilado. Por ello, algunos comandos tanto del BASIC ROM como del LASER BASIC, no pueden emplearse. En la versión ROM son: **CONT, LIST, LOAD, NEW, SAVE** y **VERIFY.** En la versión LARER son, además: **AUTO, DIR, DLOAD, DSAVE, OLD** y **RENUM.**

Compilación de programas extensos

Cuando la longitud del programa fuente supera los 20 Kbytes, no es posible cargarlo enteramente en memoria para su compilación. Para poder trabajar en estos casos, el compilador se sirve de ficheros temporales almacenados en cinta o disco. Este fichero temporal, llamado COM.TMP, se reutiliza automáticamente cada vez que es necesario, en la versión de disco.

Compilación en modo entero

Dentro de la aritmética entera, el rango de almacenamiento de valores numéricos va desde —8388608 a 8388607. En general, podemos decir que el funcionamiento de esta aritmética es el mismo que el habitual en el sistema de coma flotante, con redondeo al entero anterior por defecto.

Sin embargo, existe alguna excepción a esta regla, como es el caso de la división con resultado negativo. En este caso, se redondea el resultado al entero superior.

Podemos ver más claramente este efecto con un ejemplo:

```
10 PRINT 5*4/2
20 PRINT 5/2*4
```

El resultado de estas dos operaciones, dentro del BASIC intérprete (coma flotante), es el mismo: 10. Sin embargo, dentro de la compilación con aritmética entera, el resultado es 10 y 8, como fruto del redondeo a enteros de las operaciones intermedias.

En cuanto a la generación de aleatorios, debemos decir que su empleo es posible, respetando la norma de hacer seguir la operación **RND(1)** de la multiplicación por un número, caso por otro lado habitual en la mayoría de ocasiones, debido a que la función **RND** genera tan sólo un número de coma flotante en el intervalo: 0<=n<1.

Por último, falta decir que no está autorizado el empleo de funciones trascendentales: **LOG, EXP, COS, SIN, TAN,** y **ATN.**



Turbo para cinta

Los programas absolutos generados por el compilador, pueden leer y grabar ficheros en cinta en el formato TURBO (alta velocidad). Los programas almacenados en este formato en cinta, pueden cargarse normalmente, aunque para grabar de memoria a casete es necesario utilizar el programa de utilidad suministrado junto con el compilador.

El generador boot

Esta utilidad se suministra únicamente en la versión en disco, permitiendo cargar y ejecutar programas directamente, en la forma:

LOAD "nombre-fichero",8,1

Para ello, se almacena un programa en disco con el nombre que indiquemos, el cual se encarga de llevar a memoria el programa objeto y ponerlo inmediatamente en funcionamiento.

Uso combinado con rutinas en código máquina

Cuando se ejecuta un programa compilado, la ROM del BASIC es commutada fuera del banco de memoria accesible, y se utilizan las posiciones de memoria hasta $CFFF. Para reservar memoria, destinada a albergar rutinas en código máquina, pueden efectuarse **POKEs** a las direcciones de página cero 45 y 46 para, a continuación, ejecutar un **CLR,** de forma similar a como se trabaja en el modo intérprete.

Por otra parte, en los programas LASER BASIC existen 256 bytes disponibles, comenzando en la posición $0500, que pueden emplearse para albergar estas rutinas.

El comando **LOAD** no puede utilizarse directamente dentro de los programas compilados. La mejor forma para salvar este problema es acceder directamente a la rutina del KERNAL "LOAD" contenida en la ROM, a través del comando **SYS,** utilizando las posiciones de memoria 780 a 783 para activar los registros del microprocesador.

Los detalles sobre cómo realizar esta operación, pueden encontrarse en la Guía de Referencia del Programador del COMMODORE 64.

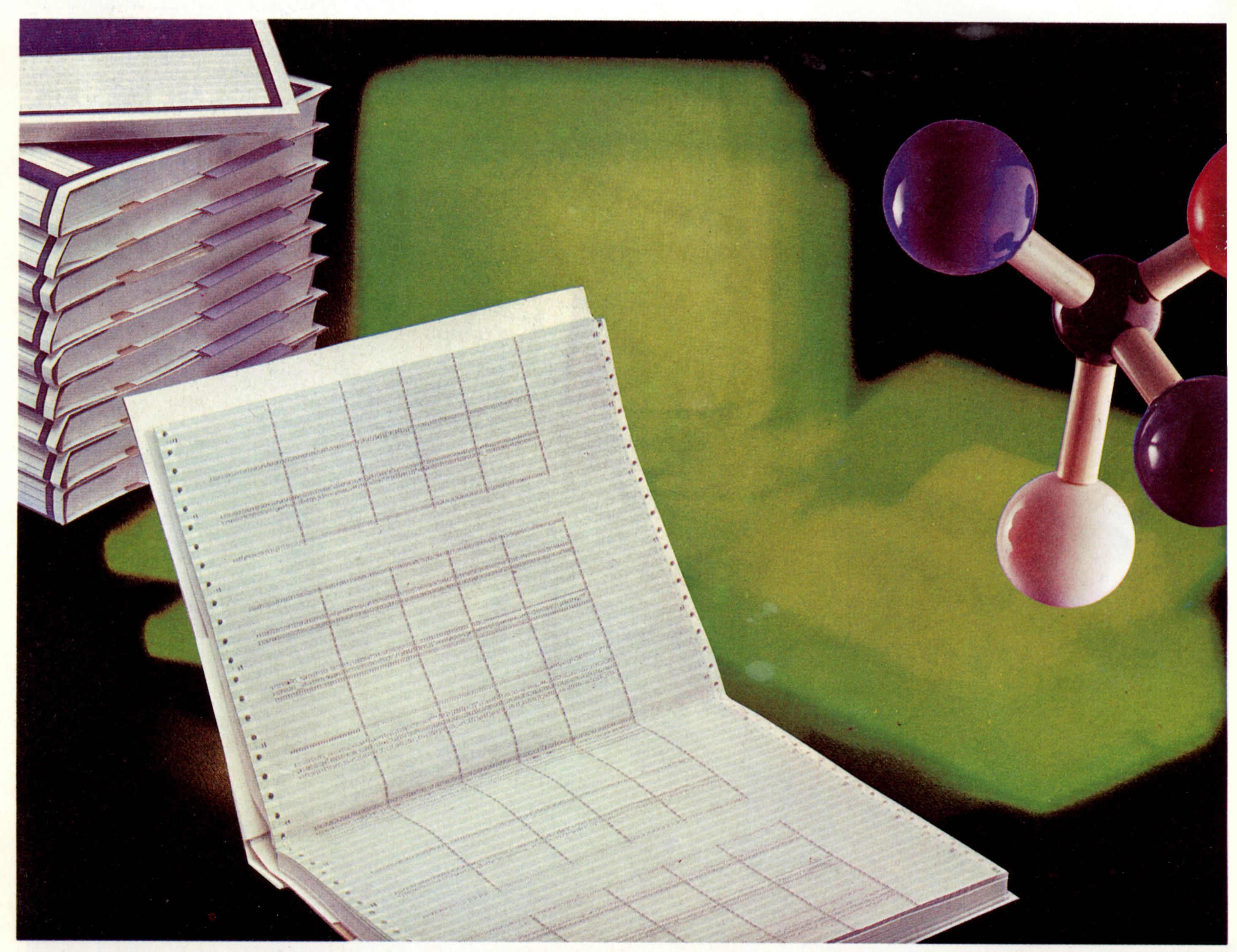
Encadenamiento de programas

Esta opción es únicamente aplicable a la versión de disco. Con ella, es posible cargar otro programa al finalizar la ejecución del contenido en memoria, ejecutándose éste de forma automática.

Este efecto se logra incluyendo en el programa fuente una sentencia **END** seguida por dos puntos y el nombre del programa a encadenar entre comillas:

END: "nombre-programa"

Nombre: Laser Compiler
Precio: 6.500 ptas. (disco)
y 5.200 ptas. (cinta)
Distribuidor: COMPULAND
Calvo Asensio, 8
Tel.: 243 16 38
28015 MADRID



El bombero

Amigo y enemigo del hombre a un mismo tiempo, el fuego y sus devastadoras consecuencias producen múltiples tragedias como podemos comprobar muy a menudo en las primeras páginas de los diarios. Gracias al valor y arrojo de las dotaciones de bomberos no todo está perdido cuando se declara un incendio.

Día 27 de febrero. Hora: 10 de la mañana. Ignorando la expresa prohibición de su madre, Juan Alonso (un niño de 8 años, inquilino del segundo piso del inmueble siniestrado) estaba jugando con fuego, en sentido literal, concretamente con una caja de cerillas. A las 10:05 el salón de la vivienda se había convertido en un horno; afortunadamente, el pequeño Juan pudo escapar.

Hora: 10:10. Lugar: Acuartelamiento de la primera brigada de bomberos. En centralita contestan a la urgente llamada telefónica concerniente al incendio que acaba de declararse. La actividad entonces es febril; todos los bomberos de guardia se preparan para salir y uno tras otro se deslizan por la barra de emergencia hasta llegar a su vehículo. Cinco minutos después de recibir la llamada, las sirenas del transporte de bomberos emiten la señal de prioridad de tráfico, mientras éste se precipita por las calles vertiginosamente.

Hora: 10:25. El vehículo frena ruidosamente frente a la fachada en llamas del inmueble. Como un solo hombre, todos los ocupantes del mismo descienden a la vez: cada uno conoce ya su cometido específico. El jefe de bomberos examina rápidamente la situación, pregunta a los vecinos acerca de los ocupantes de los diferentes pisos y empieza a dar órdenes, que sus subordinados acatan con efectividad y sin



discusión, confiando en la experiencia del mando.

Como resultado de esta labor conjunta, los primeros chorros de agua lamen las ventanas de las plantas inferiores. Al parecer todos los inquilinos están a salvo, en la calle, excepto un grupo de niños, copados por el fuego en el interior de una guardería sita casi en la azotea. ¡Una guardería! Rápido, no podemos perder tiempo... que suban algunos voluntarios, es preciso salvar a esas criaturas —grita el jefe de bomberos.

Aún a riesgo de perder tu propia vida atraviesas la puerta, entrando así en el rellano de la escalera. Desde donde te encuentras puedes oír claramente el llanto de algún bebé de la guardería. Aunque el incendio está controlado, no ha sido extinguido todavía, por eso encuentras voraces lenguas de fuego frente a ti. Sólo los reducidos depósitos de agua adheridos a tu espalda te permiten traspasar la cortina de llamas. Sujetas a uno de los bebés entre tus brazos y bajas rápidamente. Ya está a salvo, pero todavía hay otros en peligro...

Una heroica labor

Por suerte se trata exclusivamente de un relato ficticio, pero su desenlace depende directamente de ti ¿Cómo? Muy fácil. Necesitas tan sólo dos brazos firmes para la heroica labor de copiar el listado adjunto. Lo demás, un Commodore 64 o 128, y un joystick conectado al port 2.

El juego comienza, tras el consabido RUN y unos segundos de espera, con una representación esquemática del edificio donde transcurren los hechos antes descritos. En el piso bajo, a la derecha, vemos la figura inmóvil de un bombero que está sujeto al control del joystick. Si dirigimos el joystick hacia la derecha o hacia la izquierda, el personaje se moverá en dicha dirección.

En la pantalla observamos también la presencia de cuatro escaleras situadas en los extremos de la misma. Subiendo o bajando por estas escaleras, iremos cambiando de piso hasta llegar a la guardería.

Ascender entraña más dificultad que el descenso, pues en este último caso basta con situar al bombero lo más alejado posible del centro de la pantalla y empujar del joystick hacia abajo. Por el contrario, para acceder a una planta más alta es preciso situar a nuestro personaje junto a la escalera y dirigir el joystick hacia arriba. Hasta que logremos acostumbrarnos nos costará bastante trabajo dar con el lugar exacto.

Las llamas, que se han adueñado del edificio, se interponen en nuestro camino. Dos soluciones para evitar las quemaduras del fuego: esquivar las llamas o rociarlas con agua. Disponemos de una pequeña reserva de agua dentro de los depósitos que transporta consigo el bombero, depósitos que sólo puede recargar en el lavabo de la guardería. Gracias al botón de disparo del joystick cabe la posibilidad de apagar las llamas.

Para rescatar a un niño, debemos llegar hasta la guardería y descender después, con el niño en brazos, hasta el nivel inferior. El llanto de éste nos indica cuándo transportamos al niño y cuándo lo hemos dejado en brazos de un compañero, sano y salvo. Después de cada rescate, el siguiente, y el siguiente...

En total, son tres los bomberos encargados de recoger a todos los niños, de forma que si uno de ellos perece abrasado, quedan los demás. Muertos los tres, la ciudad te concederá una mención honorífica si has salvado a un cierto número de niños (cinco en un principio), en caso contrario nadie recordará tu hazaña. De nuevo, vuelta a empezar.

El programa está dotado de dificultad progresiva; es decir, conforme rescatemos a más y más niños nos resultará más difícil salvar a otros (aunque no lo parezca). No es nada personal, que conste.

LISTADO

```
10 POKE 53269,0                                    -053-
200 FOR J=0 TO 448:READ A:POKE 12288+J,A:NEXT J
                                                   -253-
210 FOR J=0 TO 50:READ A:POKE 50000+J,A:NEXT J
                                                   -179-
220 FOR J=0 TO 200:READ A:POKE 50500+J,A:NEXT J
                                                   -230-
230 FOR J=0 TO 125:READ A:POKE 51000+J,A:NEXT J
                                                   -233-
240 FOR J=0 TO 329:READ A:POKE 51500+J,A:NEXT J
                                                   -245-
250 FOR J=0 TO 194:READ A:POKE 49201+J,A:NEXT J:T=5
                                                   -251-
500 POKE 53276,255                                 -211-
510 POKE 53285,0:POKE 53286,7                      -092-
520 FOR J=1 TO 7:POKE 53287+J,2:NEXT J             -216-
530 POKE 53287,6:POKE 50549,66                     -154-
540 POKE 2040,196:POKE 2047,195                    -187-
550 FOR J=1 TO 6:POKE 2040+J,197+(J AND 1):NEXT J
                                                   -016-
560 POKE 2,8:POKE 49180,0:D=8                      -126-
570 POKE 53275,126:B=3                             -194-
5000 POKE 53265,PEEK(53265) AND 239:POKE 53281,14:POK
E 53280,6:PRINT"{CLR}{BLK}";:                      -062-
5010 PRINT"{RON}{GR1}[2 ESP]RESCATES:[16 ESP]BOMBEROS
:[4 ESP]";                                         -130-
5020 PRINT"{RON}{NRJ}[13 ESP]{ROF}"                -125-
5030 PRINT"{RON}{NRJ}[2 ESP]{BLK}GUARDERIA{NRJ}[2 ESP
]{ROF}{BLU} !BUAA!"                                -157-
5040 PRINT"{RON}{NRJ}[13 ESP]{ROF}"                -127-
5050 PRINT"{RON}{NRJ}[13 ESP]{ROF}{BLK}<27 P>{3 ARB}"
:FOR N=1 TO 2                                      -074-
5060 PRINT:PRINTTAB(38)"{ABJ}{BLK}<BLK>{RON}{CYN} {RO
F}"                                                -212-
5070 PRINTTAB(37)"{ARB}{BLK}<BLK>{RON}{CYN}[2 ESP]{RO
F}"                                                -173-
5080 PRINTTAB(36)"{ARB}{BLK}<BLK>{RON}{CYN}[3 ESP]{RO
F}"                                                -205-
5090 PRINTTAB(35)"{ARB}{BLK}<BLK>{RON}{CYN}[4 ESP]{RO
F}"                                                -237-
5100 PRINT"{ARB}{BLK}<34 P><P>{RON}{CYN}[5 ESP]";
                                                   -092-
5110 PRINT"{RON}{CYN} {ROF}{BLK}¦L¦"               -119-
5120 PRINT"{RON}{CYN}[2 ESP]{ROF}{BLK}¦L¦"         -152-
5130 PRINT"{RON}{CYN}[3 ESP]{ROF}{BLK}¦L¦"         -185-
5140 PRINT"{RON}{CYN}[4 ESP]{ROF}{BLK}¦L¦{ACL}"
                                                   -116-
5150 PRINT"{RON}{CYN}[5 ESP]{ROF}{BLK}¦L¦<33 P>{ARB}"
;                                                  -086-
5160 NEXT N                                        -153-
5165 PRINT"{HOM}{GR1}{RON}"TAB(12)PEEK(49154):PRINT"{
HOM}{GR1}{RON}"TAB(37)B                            -033-
5170 POKE 55895,0:POKE 56295,0                     -152-
5180 PRINT"{HOM}";:POKE 1623,111:POKE 2023,111
                                                   -048-
5200 POKE 53262,240:POKE 53263,227                 -077-
5210 FOR J=0 TO 3:POKE 251+J,0:NEXT J:POKE 49154,0
                                                   -125-
5220 POKE 49155,33:POKE 49156,2:POKE 49157,21:POKE 49
158,1                                              -213-
```

```
5230 POKE 49159,1:POKE 49160,33                      -193-
5240 POKE 53269,255:POKE 49152,0:POKE 789,192
                                                     -238-
5250 FOR N=0 TO 4:POKE 53253+(2*N),67+(INT(RND(1)*4)*
40)                                                  -025-
5260 POKE 53252+(2*N),INT(RND(1)*256):NEXT N
                                                     -223-
5270 POKE 53251,227:POKE 53250,INT(RND(1)*150):POKE 5
3264,0:C=1                                           -238-
5280 POKE 53294,2                                    -163-
5900 POKE 53265,PEEK(53265) OR 16:POKE 49152,1
                                                     -028-
6000 IF PEEK(49154)=1 THEN IF C<>1 THEN GOSUB 6300
                                                     -100-
6010 IF PEEK(49154)=0 THEN C=0:PRINT"{GR1}{HOM}{RON}"
TAB(12)PEEK(49153)                                   -109-
6015 P=PEEK(53278):P=PEEK(53278)                     -062-
6020 IF (P AND 1) OR (P AND 128) THEN GOTO 6100
                                                     -059-
6030 GOTO 6000                                       -008-
6100 IF P AND 128 THEN 6200                          -203-
6110 FOR N=1 TO 6:IF PEEK(53249+2*N)=PEEK(53263) THEN
 GOSUB 6150                                          -144-
6120 NEXT N:GOTO 6000                                -239-
6150 IF P AND (2^N) <> 2^N THEN RETURN               -020-
6160 IF ABS(PEEK(53248+2*N)-PEEK(53262))>40 THEN RETU
RN                                                   -170-
6170 POKE 53249+(2*N),67+(INT(RND(1)*5)*40):RETURN
                                                     -040-
6200 POKE 49152,0:POKE 53294,0:B=B-1                 -221-
6210 PRINT"{HOM}{GR1}{RON}"TAB(37)B                  -074-
6220 FOR J=0 TO 100:POKE 53294,0:POKE 53294,2:NEXT J
                                                     -217-
6230 IF B=0 THEN 19000                               -178-
6240 GOTO 5000                                       -010-
6300 C=1:POKE 254,0:D=D-0.2:IF D<1 THEN D=1          -145-
6310 POKE 2,D:POKE 49180,0:RETURN                    -224-
19000 POKE 53265,PEEK(53265) AND 239:POKE 53280,2:POK
E 53281,10                                           -200-
19010 PRINT"{CLR}{2 ABJ}[2 ESP]{RON}{MRN}¦\¦[3 ESP]<*
>{ROF}{BLK} {MRN} {RON}¦\¦[3 ESP]<*>{ROF}{BLK}[2 ESP]
{RON}{MRN} <*>{ROF}{BLK} {RON}{MRN}¦\¦ {ROF} {BLK} {R
ON}{MRN}¦\¦[3 ESP]<*>{ROF}{BLK}[11 ESP]{MRN} ";
                                                     -126-
19020 PRINT"[2 ESP]{RON}{MRN} {ROF}¦\¦ <*>{RON} {ROF}
{BLK} {MRN} {RON} {ROF}¦\¦{BLK} {MRN}<*>{RON} {ROF}{B
LK}[2 ESP]{RON}{MRN}[5 ESP]{ROF} {BLK} {RON}{MRN} {RO
F}¦\¦{BLK}[2 ESP]{MRN}<*>{BLK}[11 ESP]{MRN} ";
                                                     -075-
19030 PRINT"[2 ESP]{RON}{MRN} {ROF}{BLK}[3 ESP]{MRN}
{BLK} {MRN} {RON} {ROF}{BLK}[3 ESP]{RON}{MRN} {ROF}{B
LK}[2 ESP]{RON}{MRN} {ROF}<*>{RON} {ROF}¦\¦{RON} {ROF
} {BLK} {RON}{MRN}[2 ESP]<*>{ROF}{BLK}[13 ESP]"
                                                     -047-
19040 PRINT"[2 ESP]{RON}{MRN} {ROF}{BLK} {RON}{MRN}[3
 ESP]{ROF}{BLK} {MRN} {RON}[5 ESP]{ROF}{BLK}[2 ESP]{R
ON}{MRN} {ROF}{BLK}[3 ESP]{RON}{MRN} {ROF} {BLK} {RON
}{MRN}[2 ESP]{ROF}¦\¦{BLK}[13 ESP]{MRN} ";          -254-
19050 PRINT"[2 ESP]{RON}{MRN} <*>{ROF}{BLK} {MRN} {RO
N} {ROF}{BLK} {MRN} {RON} {ROF}¦\¦{BLK} {MRN}<*>{RON}
 {ROF}{BLK}[2 ESP]{RON}{MRN} {ROF}{BLK}[3 ESP]{RON}{M
RN} {ROF} {BLK} {RON}{MRN} <*>{ROF}{BLK}[2 ESP]{RON}{
MRN}¦\¦{ROF}{BLK}[5 ESP]"                           -167-
19060 PRINT"[2 ESP]{MRN}<*>{RON}[3 ESP]{ROF}¦\¦{BLK}
{MRN} {RON} {ROF}{BLK}[3 ESP]{RON}{MRN} {ROF}{BLK}[2
ESP]{RON}{MRN} {ROF}{BLK}[3 ESP]{RON}{MRN} {ROF} {BLK
} {MRN}<*>{RON}[3 ESP]{ROF}¦\¦{BLK}[11 ESP]{MRN} ";
                                                     -007-
19070 PRINT"{2 ABJ}[7 ESP]{MRN}[2 ESP]{BLK}[2 ESP]{MR
N} {RON}¦\¦[3 ESP]<*>{ROF}{BLK}[2 ESP]{RON}{MRN} {ROF
}{BLK}[3 ESP]{RON}{MRN} {ROF}{BLK}[2 ESP]{RON}{MRN}¦\
¦[3 ESP]<*>{ROF}[2 ESP]{RON}[4 ESP]<*>{ROF}{BLK}[2 ES
P]";                                                 -139-
19080 PRINT"[7 ESP]{MRN}[2 ESP]{BLK}[2 ESP]{MRN} {RON
} {ROF}¦\¦{BLK} {MRN}<*>{RON} {ROF}{BLK}[2 ESP]{RON}{
MRN} {ROF}{BLK}[3 ESP]{RON}{MRN} {ROF}{BLK}[2 ESP]{RO
N}{MRN} {ROF}¦\¦{BLK}[2 ESP]{MRN}<*>{BLK}[2 ESP]{RON}
{MRN} {ROF}¦\¦{BLK} {MRN}<*>";                      -059-
19090 PRINT"{RON} {ROF}{BLK}[2 ESP]";                -195-
19100 PRINT"[7 ESP]{MRN}[2 ESP]{BLK}[2 ESP]{MRN} {RON
} {ROF}{BLK}[3 ESP]{RON}{MRN} {ROF}{BLK}[2 ESP]{RON}{
MRN} {ROF}{BLK}[3 ESP]{RON}{MRN} {ROF}{BLK}[2 ESP]{RO
N}{MRN}[2 ESP]<*>{ROF}{BLK}[4 ESP]{RON}{MRN} {ROF}{BL
K}[2 ESP]{RON}{MRN}¦\¦";                            -145-
19110 PRINT"{ROF}¦\¦{BLK}[2 ESP]";                   -051-
19120 PRINT"[7 ESP]{MRN}[2 ESP]{BLK}[2 ESP]{MRN} {RON
} {ROF}{BLK}[3 ESP]{RON}{MRN} {ROF}{BLK}[2 ESP]{RON}{
MRN} {ROF}{BLK}[3 ESP]{RON}{MRN} {ROF}{BLK}[2 ESP]{RO
N}{MRN}[2 ESP]{ROF}¦\¦{BLK}[4 ESP]{RON}{MRN}[4 ESP]{R
OF}{BLK}[2 ESP]";                                    -109-
19130 PRINT"{MRN} ";                                 -223-
19140 PRINT"[7 ESP]{MRN}[2 ESP]{BLK}[2 ESP]{MRN} {RON
} <*>{ROF}{BLK} {RON}{MRN}¦\¦ {ROF}{BLK}[2 ESP]{RON}{
MRN} <*>{ROF}{BLK} {RON}{MRN}¦\¦ {ROF}{BLK}[2 ESP]{RO
N}{MRN} <*>{ROF}{BLK}[2 ESP]{RON}{MRN}¦\¦{ROF}{BLK}[2
 ESP]{RON}{MRN} {ROF}¦\¦ ";                         -032-
19150 PRINT"<*>{RON}<*>{ROF}{BLK}[2 ESP]";           -094-
19160 PRINT"[7 ESP]{MRN}[2 ESP]{BLK}[2 ESP]{MRN} <*>{
RON}[3 ESP]{ROF}¦\¦{BLK}[2 ESP]{MRN}<*>{RON}[3 ESP]{R
OF}¦\¦{BLK}[2 ESP]{MRN}<*>{RON}[3 ESP]{ROF}¦\¦{BLK}[2
 ESP]{RON}{MRN} {ROF}{BLK}[2 ESP]{MRN} {RON} {ROF}{BL
K} {BLK}"                                            -076-
19170 POKE 53264,2                                   -211-
19180 POKE 53250,40:POKE 53251,135:POKE 53252,50:POKE
 53253,70                                            -136-
19190 POKE 53254,110:POKE 53255,94:POKE 53256,148
                                                     -177-
19200 POKE 53257,85:POKE 53258,132:POKE 53259,152
                                                     -177-
19210 POKE 53260,177:POKE 53261,150                  -131-
19220 POKE 53262,164:POKE 53263,200:POKE 53294,2:POKE
 53269,254                                           -199-
19230 POKE 2047,194:P=PEEK(49153):IF P<=T THEN 19300
                                                     -186-
19240 PRINT"{BLK}{ABJ} {RON}[10 ESP]MENCION HONORIFIC
A[10 ESP]{ROF}"                                      -225-
19250 T=P:PRINT"{BLU}{4 ABJ}{RON}"TAB(9)T"{IZQ}[2 ESP
]PERSONAS SALVADAS "                                 -082-
19300 POKE 53265,PEEK(53265) OR 16                   -096-
19310 FOR N=0 TO 250:SYS 50000:FOR J=1 TO 10:NEXT J,N
:GOTO 500                                            -106-
20000 DATA 0,60,0,0,252,0,0,255,0,0,213,0,0,212,0,0
                                                     -182-
20010 DATA 168,0,0,168,0,0,152,0,0,152,0,0,153,92,0,1
64                                                   -158-
20020 DATA 0,0,168,0,0,252,0,0,168,0,0,106,0,0,154,12
8                                                    -101-
20030 DATA 2,166,144,2,129,80,1,65,64,1,80,0,1,80,0,0
                                                     -073-
20040 DATA 0,240,0,3,240,0,3,252,0,3,84,0,3,80,0,2
                                                     -155-
20050 DATA 160,0,2,160,0,2,96,0,2,96,0,2,101,112,2,14
4                                                    -106-
20060 DATA 0,2,160,0,3,240,0,106,160,0,106,160,0,90,1
60,0                                                 -244-
20070 DATA 66,128,0,66,128,0,1,128,0,1,80,0,1,80,0,0
                                                     -019-
20080 DATA 0,60,0,0,63,0,0,255,0,0,87,0,0,23,0,0
                                                     -055-
20090 DATA 42,0,0,42,0,0,38,0,0,38,0,53,102,0,0,26
                                                     -158-
20100 DATA 0,0,42,0,0,63,0,0,42,0,0,169,0,2,166,0
                                                     -101-
20110 DATA 6,154,128,5,66,128,1,65,64,0,5,64,0,5,64,0
                                                     -093-
20120 DATA 0,15,0,0,15,192,0,63,192,0,21,192,0,5,192,
0                                                    -112-
20130 DATA 10,128,0,10,128,0,9,128,0,9,128,13,89,128,
0,6                                                  -234-
20140 DATA 128,0,10,128,0,15,192,0,10,169,0,10,169,0,
10,165                                               -109-
20150 DATA 0,2,129,0,2,129,0,2,64,0,5,64,0,5,64,0
                                                     -124-
20160 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0           -231-
20170 DATA 0,0,0,6,0,0,42,0,1,169,128,42,154,168,0,16
9                                                    -127-
20180 DATA 128,0,42,0,0,2,0,0,0,0,0,0,0,0,0,0
                                                     -140-
```

```
20190 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0        -234-
20200 DATA 0,0,0,0,0,0,0,0,0,2,128,0,2,160,0,0
                                                   -184-
20210 DATA 168,0,0,170,0,0,170,0,0,174,0,2,174,128,2,
170                                                -209-
20220 DATA 128,2,170,160,2,170,160,2,186,160,2,187,14
4,10,175,208                                       -172-
20230 DATA 10,239,64,10,191,64,10,185,0,6,180,0,1,80,
0,0                                                -219-
20240 DATA 0,0,0,0,0,0,0,2,0,0,8,0,0,8,0,0        -248-
20250 DATA 40,0,0,40,0,0,40,0,0,42,0,0,170,128,2,170
                                                   -246-
20260 DATA 128,2,170,160,2,170,160,2,186,160,2,187,14
4,10,191,208                                       -174-
20270 DATA 10,255,64,10,251,64,11,235,128,7,254,0,1,8
8,0,0                                              -077-
20280 DATA 78                                      -197-
21000 DATA 166,251,232,134,251,224,15,240,1,96,173,24
9,7,201,197,208                                    -080-
21010 DATA 7,169,198,160,197,76,108,195,169,197,160,1
98,141,249,7,140                                   -179-
21020 DATA 250,7,141,251,7,140,252,7,141,253,7,140,25
4,7,162,0                                          -018-
21030 DATA 134,251,96                              -071-
30000 DATA 173,0,220,201,119,208,48,173,16,208,41,128
,201,128,240,29                                    -063-
30010 DATA 174,14,208,224,254,208,8,173,16,208,9,128,
141,16,208,232                                     -032-
30020 DATA 232,234,142,14,208,32,179,197,96,234,234,2
34,234,174,14,208                                  -190-
30030 DATA 224,62,208,235,76,56,199,173,0,220,201,123
,208,113,173,16                                    -076-
30040 DATA 208,41,128,201,128,208,27,174,14,208,224,0
,208,8,173,16                                      -231-
30050 DATA 208,41,127,141,16,208,202,202,234,142,14,2
08,32,211,197,96                                   -118-
30060 DATA 234,234,174,14,208,224,22,208,237,234,234,
234,96,234,96,166                                  -197-
30070 DATA 252,232,134,252,224,5,240,1,96,162,0,134,2
52,174,255,7                                       -181-
30080 DATA 224,192,240,5,162,192,76,207,197,162,193,1
42,255,7,96,166                                    -112-
30090 DATA 252,232,134,252,224,5,240,1,96,162,0,134,2
52,174,255,7                                       -183-
30100 DATA 224,194,240,5,162,194,76,239,197,162,195,1
42,255,7,96,173                                    -114-
30110 DATA 0,220,41,16,201,0,240,1,96,166,253,224,1,2
08,5,162                                           -212-
30120 DATA 0,76,10,198,162,1,134,253,96            -187-
40000 DATA 166,253,224,1,240,9,173,21,208,41,254,141,
21,208,96,166                                      -232-
40010 DATA 254,224,255,208,7,162,0,134,253,76,62,199,
173,16,208,41                                      -247-
40020 DATA 128,201,128,240,240,162,20,232,232,224,112
,240,8,236,14,208                                  -150-
40030 DATA 208,245,76,77,199,162,232,232,232,224,254,
240,8,236,14,208                                   -145-
40040 DATA 208,245,76,77,199,166,254,232,134,254,173,
14,208,174,255,7                                   -164-
40050 DATA 224,192,208,6,24,105,20,76,164,199,224,193
,240,246,224,194                                   -145-
40060 DATA 208,6,56,233,20,76,164,199,224,195,240,246
,141,0,208,173                                     -044-
40070 DATA 15,208,141,1,208,173,21,208,9,1,141,21,208
,96                                                -246-
50000 DATA 173,0,220,201,126,208,55,174,14,208,172,15
,208,173,16,208                                    -067-
50010 DATA 41,128,201,128,240,20,192,227,208,8,224,62
,208,3,32,174                                      -225-
50020 DATA 201,96,192,147,240,244,96.234,234,234,192,
187,208,8,224,26                                   -151-
50030 DATA 208,3,32,224,201,96,192,107,240,244,96,234
,234,234,173,0                                     -027-
50040 DATA 220,201,125,240,1,96,174,14,208,172,15,208
,173,16,208,41                                     -018-
50050 DATA 128,201,128,240,20,192,187,208,8,224,22,20
8,3,32,18,202                                      -226-
50060 DATA 96,192,107,240,244,96,234,234,234,192,147,
208,8,224,66,208                                   -158-
50070 DATA 3,32,68,202,96,192,67,240,244,96,234,234,2
34,234,234,234                                     -049-
50080 DATA 234,234,162,0,206,14,208,206,15,208,138,41
,2,201,2,208                                       -167-
50090 DATA 5,169,194,76,196,201,169,195,141,255,7,138
,162,0,160,0                                       -211-
50100 DATA 200,192,255,208,251,232,224,20,208,244,170
,232,224,40,208,212                                -255-
50110 DATA 96,234,234,234,162,0,238,14,208,206,15,208
,138,41,2,201                                      -226-
50120 DATA 2,208,5,169,192,76,246,201,169,193,141,255
,7,138,162,0                                       -202-
50130 DATA 160,0,200,192,255,208,251,232,224,20,208,2
44,170,232,224,40                                  -154-
50140 DATA 208,212,96,234,234,234,162,0,238,14,208,23
8,15,208,138,41                                    -084-
50150 DATA 2,201,2,208,5,169,192,76,40,202,169,193,14
1,255,7,138                                        -146-
50160 DATA 162,0,160,0,200,192,255,208,251,232,224,20
,208,244,170,232                                   -106-
50170 DATA 224,40,208,212,96,234,234,234,162,0,206,14
,208,238,15,208                                    -077-
50180 DATA 138,41,2,201,2,208,5,169,194,76,90,202,169
,195,141,255                                       -204-
50190 DATA 7,138,162,0,160,0,200,192,255,208,251,232,
224,20,208,244                                     -017-
50200 DATA 170,232,224,40,208,212,96,234,234,234
                                                   -113-
60000 DATA 173,0,192,201,1,240,3,76,49,234,32,80,195,
32,68,197                                          -043-
60010 DATA 32,56,199,32,44,201,234,234,234,234,234,23
4,234,234,234,234                                  -178-
60020 DATA 234,234,234,234,234,234,234,234,234,234,23
4,234,234,234,234,234                              -118-
60030 DATA 234,234,234,234,234,234,234,174,14,208,172
,15,208,224,126,208                                -022-
60040 DATA 27,192,67,208,20,169,1,141,2,192,162,0,169
,32,157,94                                         -099-
60050 DATA 4,232,224,6,208,248,174,14,208,76,189,192,
224,66,208,249                                     -061-
60060 DATA 192,227,208,245,173,16,208,41,128,201,128,
208,236,173,2,192                                  -190-
60070 DATA 201,1,208,229,169,0,141,2,192,238,1,192,16
2,0,189,3                                          -034-
60080 DATA 192,157,94,4,232,224,6,208,245,234,234,234
,174,28,192,232                                    -100-
60090 DATA 142,28,192,228,2,240,3,76,49,234,162,0,142
,28,192,238                                        -151-
60100 DATA 2,208,238,4,208,238,4,208,238,6,208,238,6,
208,206,8                                          -049-
60110 DATA 208,206,10,208,206,10,208,206,12,208,206,1
2,208,206,12,208                                   -101-
60120 DATA 76,49,234                               -030-

READY.                                             -173-
```

Un fre() como debe ser

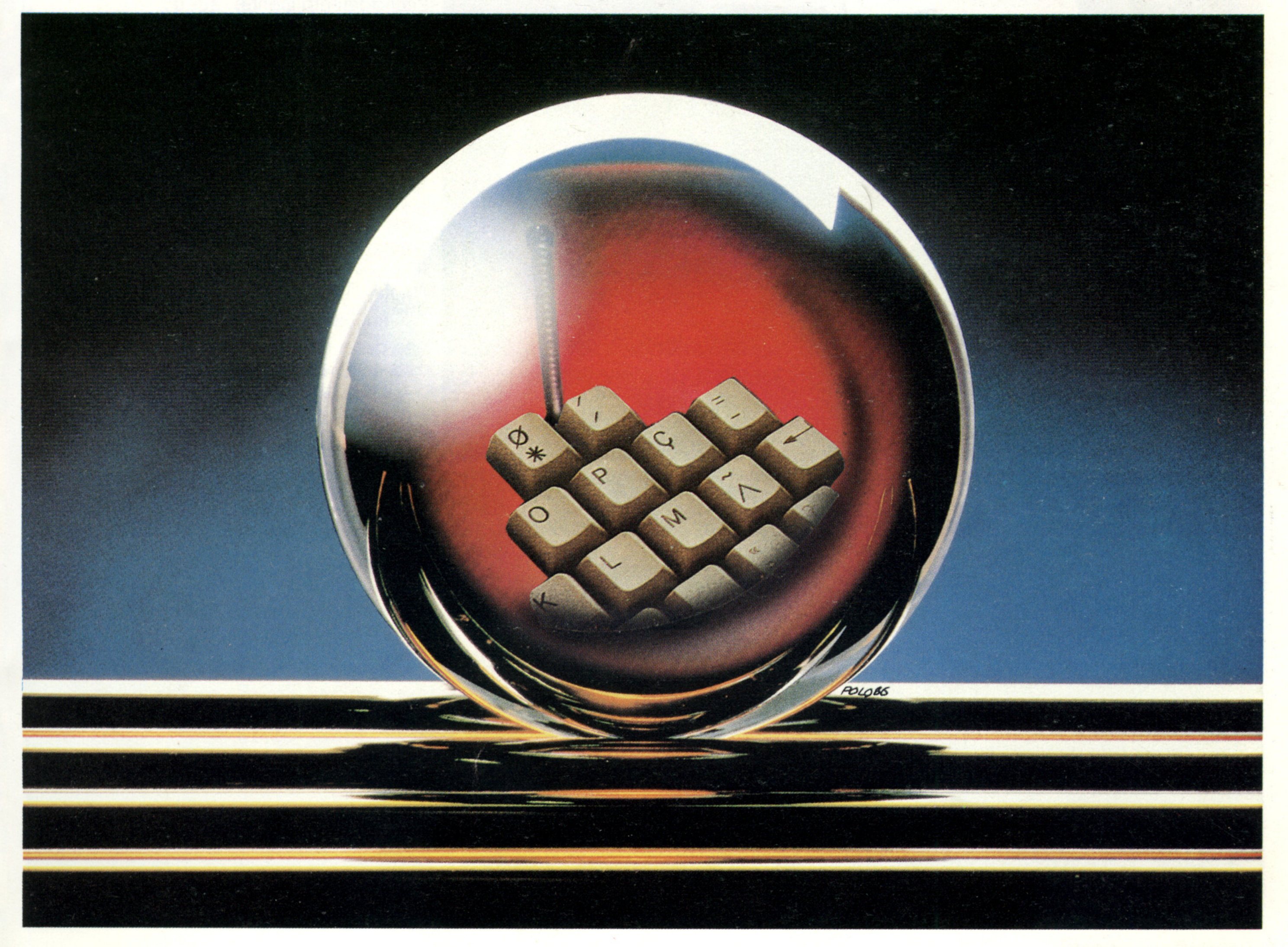
Cómo saber el tamaño real de los programas en BASIC

El C-64 es un ordenador muy flexible para introducir rutinas propias del usuario, interceptando a las ya existentes. Un ejemplo analizado ampliamente con anterioridad es la intercepción de la rutina IRQ. Pero, como comprobaremos este mes, existen muchas otras rutinas a nuestra disposición.

Seguro que alguna vez has sentido curiosidad por saber exactamente el tamaño de tus programas BASIC. El comando **FRE ()**, nos puede orientar ligeramente, ya que su resultado no es el tamaño del programa BASIC, sino la cantidad de memoria libre para éste; además, si hemos utilizado variables, éstas restarán espacio al BASIC, por si fuera poco, deberemos conocer con anterioridad el total de memoria libre para poder efectuar la resta y por último, si la cantidad de memoria libre supera cierta cantidad, el resultado de **FRE()** será un número negativo complementario. Total: esta función no nos sirve para nuestro propósito.

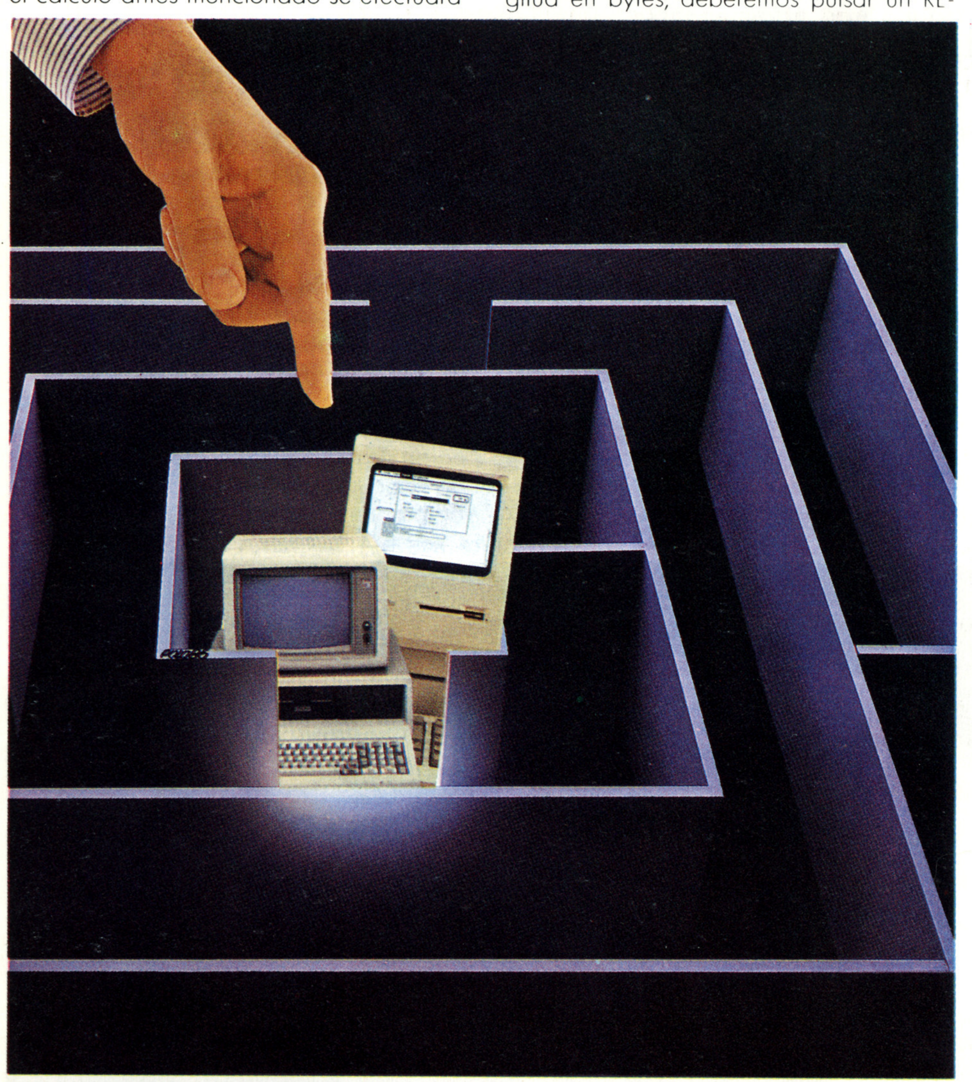
Para conocer el tamaño en bytes de un programa BASIC, deberemos acudir a los punteros de inicio del BASIC (43-44 decimal), y al de final del mismo (45-46 decimal), quedando el cálculo como sigue: Programa=256*(PEEK(46)-PEEK(44))+(PEEK(45)-PEEK(43)).

Liosillo ¿no? Para eso, hemos creado el programa del listado adjunto, que se encarga de efectuar el engorroso cálculo cada vez que pulsamos RETURN a una línea de programa (o a cualquier entrada).



Su uso es muy sencillo, y opera de una manera análoga al ¡Fuera Errores!, por lo cual, ambos programas son incompatibles. Tecleemos el programa del listado y grabémoslo antes de ejecutarlo. Cuando esté listo, pulsemos RUN y veremos aparecer unos mensajes en la pantalla sobre su uso; a partir de ahí, cada vez que pulsemos RETURN, el cálculo antes mencionado se efectuará automáticamente y aparecerá en la esquina superior izquierda de la pantalla (como en el Fuera Errores!), un mensaje en reverso indicando la cantidad en bytes que ocupa el programa en memoria.

Como el cálculo se efectúa antes del RETURN, cuando terminemos de teclear el programa del que queramos saber su longitud en bytes, deberemos pulsar un RETURN «en vacío», con lo que aparecerá el número correcto, incluida la última línea introducida. Así pues, para averiguar la longitud de un programa no tecleado en el momento, lo único que hay que hacer es cargarlo, y pulsar RETURN.



Funcionamiento

El programa BASIC no es más que un cargador del código máquina necesario para su funcionamiento, que se ubicará en la tan socorrida área de memoria del buffer del casete, por lo que, si utilizamos este periférico para la carga de programas, deberemos atender a las indicaciones que aparecen siempre en nuestra revista, USO DEL CASETE CON FUERA ERRORES, teniendo en cuenta que el código máquina ocupa desde la posición 822 hasta la 944. Los usuarios de disco no tendrán este problema. Insistimos en que todo lo anterior, es sólo para saber la longitud de programas no tecleados, y no influirá para nada en los programas que se vayan tecleando, que es para lo que ha sido diseñado el programa fundamentalmente.

Como el ¡Fuera Errores!, este programa intercepta el vector de la rutina CHRIN (entrada de caracteres), desviándola provisionalmente para detectar la pulsación de la tecla RETURN, y efectuar el cálculo consabido. El puntero de este vector se encuentra en las posiciones de memoria hexadecimal $0324-$0325 (804-805 decimal), y lo primero que hace el programa de código máquina es cambiar estos valores, para dirigir la entrada de caracteres a nuestra propia rutina, y después continuar normalmente.

Por último, y para un mejor entendimiento de la rutina, podemos recurrir al listado en ensamblador que se adjunta, cuyos comentarios al final de cada línea irán esclareciendo los distintos aspectos de su funcionamiento.

LISTADO

```
10 FORI=822TO943:READA:S=S+A:POKEI,A:NEXT          -133-
20 IFS<>16257THENPRINT"{CLR}ERROR EN DATAS, USA: {RON
}FUERA ERRORES!{ROF}":END                          -090-
30 PRINT"{CLR}{RON}ACTIVAR{ROF}[3 ESP]: SYS822"
                                                   -234-
40 PRINT"{RON}DESACTIVAR{ROF}: RUN-STOP/RESTORE"
                                                   -244-
50 PRINT"{ABJ}EXTENSION DEL BASIC ACTIVADA"        -248-
60 SYS822                                          -033-
100 DATA 169,3,141,37,3,169,65,141,36,3,96,32,87,241,
133,251                                            -203-
110 DATA 134,252,132,253,8,201,13,240,8,165,251,166,2
52,164,253,40                                      -225-
120 DATA 96,169,13,32,210,255,165,214,141,176,3,206,1
76,3,169,0                                         -091-
130 DATA 133,216,169,18,32,210,255,169,19,32,210,255,
169,45,32,210                                      -238-
140 DATA 255,165,45,56,229,43,170,165,46,56,229,44,32
,205,189,169                                       -223-
150 DATA 32,32,210,255,169,66,32,210,255,169,89,32,21
0,255,169,84                                       -206-
160 DATA 32,210,255,169,69,32,210,255,169,83,32,210,2
55,169,45,32                                       -201-
170 DATA 210,255,173,176,3,133,214,76,79,3         -087-

READY.                                             -173-
```

```
DIREC HEXA.     MNEMO.    COMENTARIO
----- -------- ------    ----------

,0336 A9 03     LDA #$03 ;Cargamos el acumulador con el byte alto de la nueva
rutina
,0338 8D 25 03 STA $0325;y lo colocamos en la direccion del vector (alto)
,033B A9 41     LDA #$41 ;Cargamos el acumulador con el byte bajo de la nueva
rutina
,033D 8D 24 03 STA $0324;y lo colocamos en la direccion del vector (bajo)
,0340 60        RTS      ;Volvemos con la rutina CHRIN cambiada
,0341 20 57 F1 JSR $F157;Llamada a la rutina CHRIN
,0344 85 FB     STA $FB  ;Guardamos el Acumulador,
,0346 86 FC     STX $FC  ;el registro X,
,0348 84 FD     STY $FD  ;el registro Y
,034A 08        PHP      ;y el registro de estado
,034B C9 0D     CMP #$0D ;Comparamos con 13 (RETURN)
,034D F0 08     BEQ $0357;si RETURN vamos a $0357
,034F A5 FB     LDA $FB  ;Recuperamos el Acumulador,
,0351 A6 FC     LDX $FC  ;el registro X,
,0353 A4 FD     LDY $FD  ;el registro Y
,0355 28        PLP      ;y el registro de estado
,0356 60        RTS      ;Volvemos al BASIC
,0357 A9 0D     LDA #$0D ;Imprimimos
,0359 20 D2 FF JSR $FFD2;un RETURN (todavia no se ha hecho con la rutina
interceptada)
,035C A5 D6     LDA $D6  ;Guardamos la posicion del cursor en la linea
,035E 8D B0 03 STA $03B0;en la direccion $03B0
,0361 CE B0 03 DEC $03B0;Decrementamos en una unidad la posicion del cursor (el
RETURN no cuenta)
,0364 A9 00     LDA #$00 ;Desactivamos el
,0366 85 D8     STA $D8  ;modo insercion por si acaso estuviese activado
,0368 A9 12     LDA #$12 ;Imprimimos
,036A 20 D2 FF JSR $FFD2;Reverse ON
,036D A9 13     LDA #$13 ;Imprimimos
,036F 20 D2 FF JSR $FFD2;HOME
,0372 A9 2D     LDA #$2D ;Imprimimos
,0374 20 D2 FF JSR $FFD2;un "-"
,0377 A5 2D     LDA $2D  ;Calculamos la diferencia
,0379 38        SEC      ;de los bytes bajos de inicio
,037A E5 2B     SBC $2B  ;y final del BASIC
,037C AA        TAX      ;Traspasamos el contenido al registro X
,037D A5 2E     LDA $2E  ;Calculamos la diferencia
,037F 38        SEC      ;de los bytes altos de inicio
,0380 E5 2C     SBC $2C  ;y final del BASIC
,0382 20 CD BD JSR $BDCD;Llamamos a la rutina $BDCD que transforma el contenido
del Acumulador y del registro X en un entero de dos bytes (imprime la cantidad
en bytes del programa en memoria)
,0385 A9 20     LDA #$20 ;Imprimimos
,0387 20 D2 FF JSR $FFD2;un " "
,038A A9 42     LDA #$42 ;Imprimimos
,038C 20 D2 FF JSR $FFD2;una "B"
,038F A9 59     LDA #$59 ;Imprimimos
,0391 20 D2 FF JSR $FFD2;una "Y"
,0394 A9 54     LDA #$54 ;Imprimimos
,0396 20 D2 FF JSR $FFD2;una "T"
,0399 A9 45     LDA #$45 ;Imprimimos
,039B 20 D2 FF JSR $FFD2;una "E"
,039E A9 53     LDA #$53 ;Imprimimos
,03A0 20 D2 FF JSR $FFD2;una "S"
,03A3 A9 2D     LDA #$2D ;Imprimimos
,03A5 20 D2 FF JSR $FFD2;un "-"
,03A8 AD B0 03 LDA $03B0;Recogemos la posicion del cursor
,03AB 85 D6     STA $D6  ;La colocamos en su registro, para volver a la posicion
inicial (antes de ir a HOME para escribir los bytes)
,03AD 4C 4F 03 JMP $034F;Continuamos con el resto de la rutina.
```

Digi-View 64

La revolución de este siglo la protagoniza sin duda la evolución tecnológica de la información y las comunicaciones. Ahora esa revolución se nos «cuela» en casa a través de distintos dispositivos conectables a nuestro humilde «home computer». Si poseemos un C64, ya podemos tener nuestro digitalizador de imágenes.



omencemos por lo casi evidente:

IMAGEN: Representación de alguna cosa en pintura, escultura, dibujo, fotografía, etc.

Esta definición de imagen, o alguna parecida, es la que encontraremos en el diccionario. No obstante, actualmente se nos antoja algo obsoleta, ya que sin duda esta palabra nos evoca otras cosas como: televisión, informática, satélite, etc. Desde luego, el tratamiento de la imagen ha contribuido en gran medida a la revolución tecnológica que mencionábamos al principio del artículo, este tratamiento va, hoy día, ligado íntimamente con la Informática y, lo que hace pocos años era privilegio exclusivo de unos pocos, hoy está al alcance de todos, gracias a la evolución de la Informática del hogar.

El video-digitalizador de la casa PRINT TECHNIK que este mes vamos a ver en nuestra sección de equipos, nos abre las puertas de un mundo fascinante, que ahora podemos disfrutar en nuestro C-64: la digitalización de cualquier imagen, en la pantalla de nuestro ordenador.

La digitalización de la imagen es el proceso por el cual la señal proveniente de una fuente de captación, es convertida en una matriz de puntos representable por el ordenador. En el caso del digitalizador de imagen PRINT TECHNIK, la fuente de captación es una cámara de video del tipo blanco y negro, y el ordenador requerido es un C64 (o C128 en modo 64). Adelantaremos ya que también existe un modelo destinado al Amiga, que comentaremos en un próximo número.

El software

Como es norma habitual en muchos equipos, éstos vienen acompañados de un paquete de software específico, que gestiona su buen funcionamiento. El PRINT TECHNIK no es una excepción y necesita de ciertos programas. Estos son de una gran calidad y están diseñados con claridad para hacer que el usuario sólo se concentre en la digitalización, en vez de hacerlo en el teclado del ordenador.

Los programas se presentan en dos ver-

siones: cinta o disco. Como la unidad de disco 1541 no es precisamente un prodigio de velocidad, la versión disco, incluye en su directorio un «turbo», que el usuario no tiene que preocuparse de cargar, ya que cuando desde cualquier módulo del programa se necesite cargar algo del disco (incluido el programa principal), el «turbo» se activará automáticamente, incluso en la carga de imágenes grabadas.

Actualmente, los programas conducidos por menú, hacen un buen uso de los iconos (figuras alusivas a la tarea que realizan), y el PRINT TECHNIK no es una excepción. En la pantalla principal, aparece un menú con todas las opiniones que necesitaremos para llevar a cabo nuestra labor.

Resultados

Para digitalizar la imagen que deseemos, iluminaremos bien la persona o el objeto elegido, enfocaremos la cámara y llevaremos el cursor al icono «digitize». Al cabo de unos cinco segundos, en los que la pantalla se quedará «en blanco», volverá a aparecer el menú, y ¡ya está!, si, pero... ¿dónde? Muy sencillo, si somos de ese tipo de personas que tienen curiosidad por ver lo que ha hecho, situemos el cursor sobre el icono «view» y podremos apreciar el resultado.

¿Cómo que «qué cursor»? Bien, nos olvidábamos de decir que el paso de una opción a otra se realiza con las teclas de cursor, y que las opciones elegidas parpadean constantemente.

Ahora ya tenemos ante nosotros una imagen de nosotros mismos (no es muy original de nuestra parte); pero, parece que no nos colocamos bien delante de la cámara y la oreja derecha no ha salido en la imagen.

No importa. Hemos de saber que la digitalización se ha efectuado «de sobra», y podremos desplazar la imagen en cualquier dirección, con las familiares teclas de cursor, o ver la imagen que parte de la esquina superior izquierda con la tecla SHIFT+CLR/HOME. Esto es posible ya que una digitalización forma un cuadrado de 256x256 puntos, y como en la pantalla multicolor del C64 caben sólo 160x200, tenemos el márgen restante para un correcto centrado.

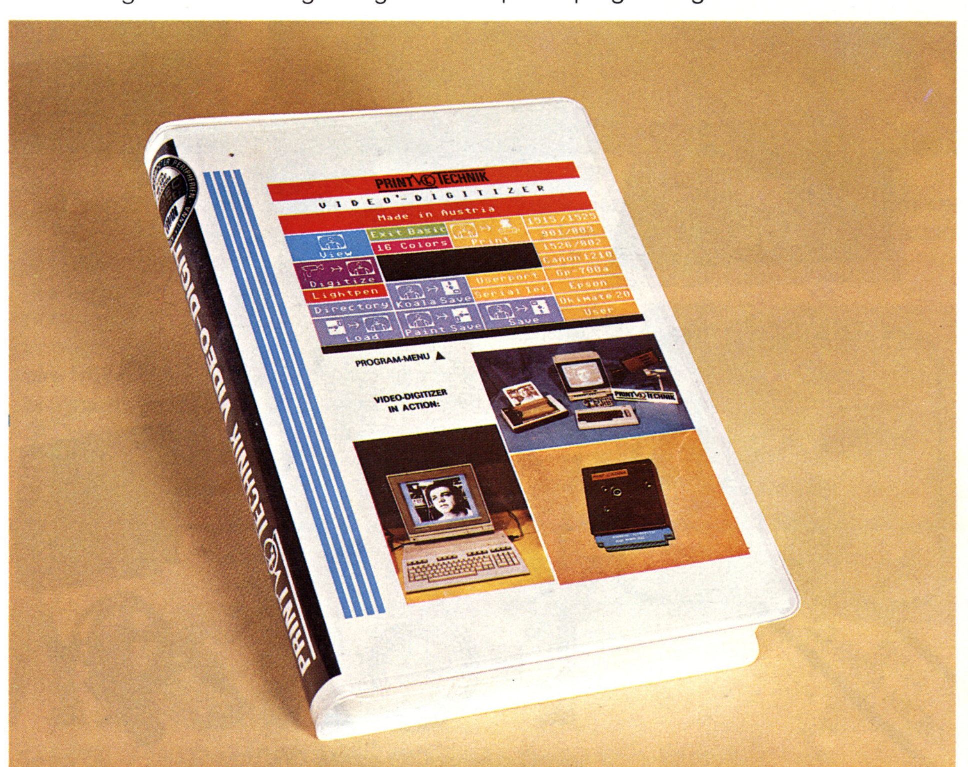
Además de poder trasladar la imagen arriba, abajo, a izquierda y a derecha, también podremos cambiar los cuatro colores por los que está formada, a saber: blanco, negro, gris 1 y gris 2. A tal fin, deberemos utilizar las cuatro teclas de función F1, F3, F5 y F7, que alteran cada una su color correspondiente, incrementando el valor del color a cada pulsación. Si por el contrario, preferimos decrementar los colores, deberemos utilizar SHIFT+ la tecla de función.

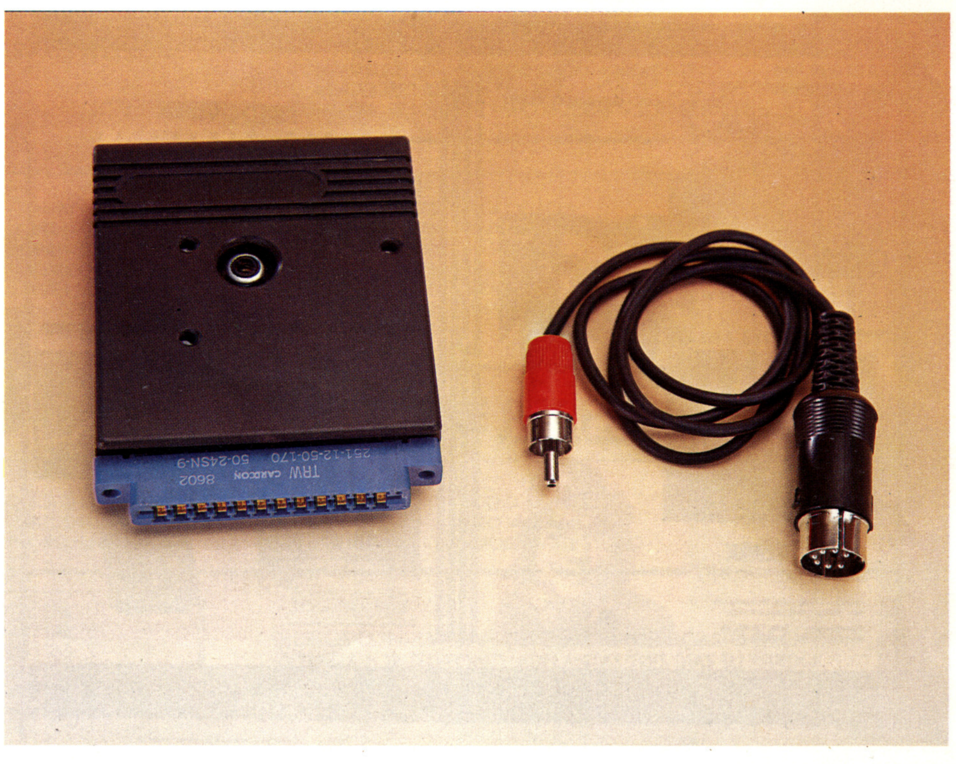
Tras ensayar con una increíble cantidad de combinaciones, lo más normal es que nos demos por vencidos y decidamos que los colores originales estaban mejor que los nuestros (la cara verde no nos favorece nada). Bien, no hay que tirar el ordenador, ni tan siquiera apagarlo y volver a empezar, basta con pulsar la tecla £ (PFOUND para los amigos alemanes), y todo volverá a sus hermosas gradaciones de gris originales.

Una vez que obtenemos la imagen correcta, lo primero que tenemos que hacer es grabarla. Para ello, el PRINT TECHNIK nos brinda tres opciones «SAVE»:

— La normal sirve para grabar los 256x256 puntos originales en formato «PRINT TECHNIK».

— La segunda grabará la imagen que haya en la pantalla en ese momento (160x200 puntos), bajo el formato utilizable por el programa gráfico KOALA.





— Por último, la tercera cumplirá una función similar a la anterior, pero bajo el formato PAINT MAGIC.

Estas dos últimas opciones, son muy interesantes para dar los últimos retoques a la imagen final, si disponemos de cualquiera de los dos programas citados. También es posible retocar la imagen con un lápiz óptico (opción «lightpen»), para lo cual deberemos disponer de una rutina especial de manejo de este dispositivo (no suministrada).

Por el contrario, para visualizar cualquier imagen grabada en disco, disponemos de la opción «load» que, lógicamente, sólo carga las imágenes grabadas en formato PRINT TECHNIK, es decir, con 256x256 puntos. Como programas separados, existen en el directorio tres, que son los siguientes:





ALARM: Sirve para comparar dos imágenes, extrayendo finalmente un resultado numérico de las diferencias entre ambas. Solicita un número inicial de comparación, que no debe ser superior a 5000 para que las diferencias computadas no sean excesivas.

VIDEO DIASHOW: Este programa se emplea para visualizar una a una, en un bucle sin fin, todas las imágenes contenidas en un disco con formato KOALA. El tiempo de visualización en segundos para cada imagen, se nos pide al principio.

VIEW: Es un programa que sirve por sí solo (como los anteriores), para visualizar una imagen grabada con formato KOALA en pantalla.

Todos los programas anteriores, disponen de un menú que incluye entre otras las opciones: Load, Save, Print, View...

Imprimiendo

Para obtener una copia impresa de nuestro trabajo, disponemos de varias ayudas en el menú. Lo primero que debemos hacer es seleccionar por dónde saldrá la información para la impresora, para lo cual existen dos opciones: la «serial IEC», que es la normal de Commodore y la más rápida; y la «userport», que al ser paralelo dará más velocidad a la impresión. Para utilizar una impresora de este tipo, habrá que cargar previamente una rutina de manejo del «userport» (no suministrada).

El tipo de impresora conectada al ordenador será indicado entre las siguientes:

— Impresoras Commodore: 1515, 1525, 801, 803, 1526 y 802).

— Impresoras no Commodore: GP-700A, Epson, Okimate 20 y User (ver opción «userport»).

Una vez seleccionado el periférico, imprimiremos la imagen en memoria con la opción del icono «print», o simplemente pulsando «P». La calidad de la misma una vez impresa es mejor aún que la de la pantalla (a nuestro parecer), y puede mejorar aún más con impresoras en color o seleccionando la opción «16 colours», que se traduce en impresión de colores en las impresoras que lo soporten, y en 16 entramados de puntos que dan, tanto en pantalla como en impresora, 16 tonalidades diferentes de grises. Esta modalidad, ocupará en disco exactamente el doble que la de cuatro colores (de 65 bloques a 135 bloques).

No obstante, obviamente, la calidad de la imagen final dependerá de varios factores, entre los que se cuentan la perfecta iluminación del objeto (brillos, sombras, etc.), la calidad de la cámara empleada y la de la impresora utilizada.





Hardware

El equipo consiste en un cartucho conectable al port del usuario, con una toma para la videocámara y tres potenciómetros ajustables con destornillador que son:

KONTRAST: Ajustamos el contraste de la imagen.

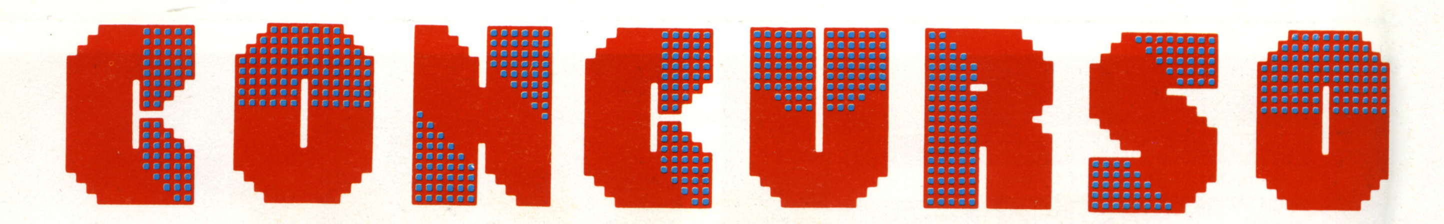
BILD BREITE: Con el que se ajusta el brillo de la imagen.

HELLIGKEIT: Ajustamos con él la luminosidad del blanco de la imagen.

Aparte de las funciones propias del digitalizador, cuenta también con dos opciones generales, una de ellas (directory) nos permite ver el directorio de un disco sin estropear el programa de la memoria, y la otra (exit BASIC), nos permite abandonar el programa, quedando la memoria de BASIC estructurada con la imagen desde $2000 hasta $4000.

En resumen, el producto nos ha parecido simplemente excelente, y se suministra con sus instrucciones oportunamente traducidas al castellano (Gracias...)

Nombre: Digi-View 64 (PRINT TECHNIK)
Precio: 42.000 ptas. (IVA incluído)
Distribuidor: COMPULOGICAL, S.A.
Sta. Cruz de Marcenado, 31
Tel.: (91) 241 10 63
28015 MADRID





El travieso C-Byte tiene el honor de invitaros a la participación en nuestro sexto concurso de programación. Los requisitos necesarios son bien pocos:

- Saber programar un ordenador COMMODORE.
- Ser español o extranjero y
- Tener una edad comprendida entre 5 y 105 años.

Fácil, ¿verdad?
En cuanto al premio, la mar de atractivo:

100.000 Ptas.

en material informático.

Y en fin, si alguno de los programas destaca por su originalidad, estética o comicidad, no sería de extrañar que le cayera alguna cosilla más...

BASES DEL CONCURSO

1. Los programas remitidos al concurso deberán ser creación original del autor o autores, y completamente inéditos, pudiendo remitir tantos programas como se desee.

2. Los p:ogramas deberán ser enviados en casete o disco flexible a TU MICRO COMMODORE (Concurso de programación). Apartado de Correos 61.294. 28080 MADRID.

3. Los programas podrán ser de cualquier tipo (juegos, utilidades, gestión, educativos) y habrán de estar escritos en lenguaje BASIC o Código Máquina.

4. Los programas deberán ser remitidos desprovistos de cualquier tipo de protección, que impida o dificulte el análisis del mismo, así como reproducción en estas páginas y su introducción como listado siguiendo el sistema FUERA ERRORES.

5. Cuando la ejecución del programa precise de la concurrencia de determinado periférico o aditamento (joysticks, tabletas gráficas, programas comerciales de ayuda), se valorará la indicación de las modificaciones pertinentes, para que el programa pueda ser disfrutado por cualquier usuario en la configuración básica.

6. Todo programa presentado al concurso deberá acompañarse de los siguientes datos:

- Datos personales del concursante.
- Nombre del programa.
- Modelo para el que está destinado.
- Breve descripción del programa detallando las indicaciones necesarias para su ejecución.

7. Los programas premiados pasarán a ser propiedad de la revista TU MICRO COMMODORE, pudiendo hacer ésta libre uso de ellos, y renunciando sus autores a cualquier otra compensación distinta al premio.

8. Los programas no premiados que por su calidad se hagan merecedores de su publicación, serán adquiridos por la editorial, aplicando la tarifa vigente.

9. Los programas recibidos con posterioridad a la fecha tope de la presente edición, serán automáticamente incluidos en los destinados a la siguiente.

10. El jurado decidirá sobre todos los aspectos no contemplados en estas bases y su decisión será inapelable.

Y ahora a darse mucha prisa, el plazo para la recepción de programas termina el próximo día 1 de Junio de 1987.

¡¡SUERTE!!



CURSO DE BASIC

SPRITES Y OTRAS CARACTERÍSTICAS GRÁFICAS

Hemos visto, hasta ahora, gran parte de las habilidades del VIC II, para facilitar las labores de creación de programas gráficos. Sin embargo, nos restan aún por comentar algunas características: las referentes a la detección de colisiones entre sprites, o entre éstos y el fondo impreso de la pantalla, así como la de transparencias de la pantalla (efecto tridimensional).

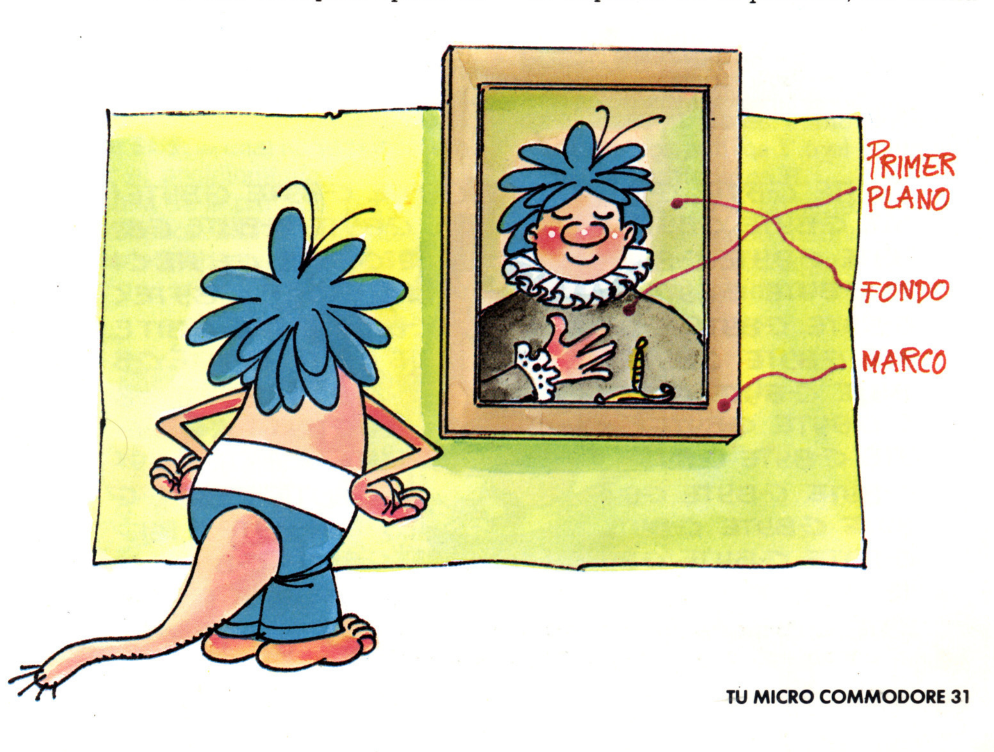
PRIORIDADES ENTRE SPRITES

Cuando hablamos de los principios básicos del color en el COMMODORE, ya tocamos el tema de la superposición de los planos. En general, podemos decir que en la pantalla existen tres planos superpuestos: el borde o marco (zona no accesible para la impresión), el fondo (zona accesible) y el primer plano (color con el que se imprime).

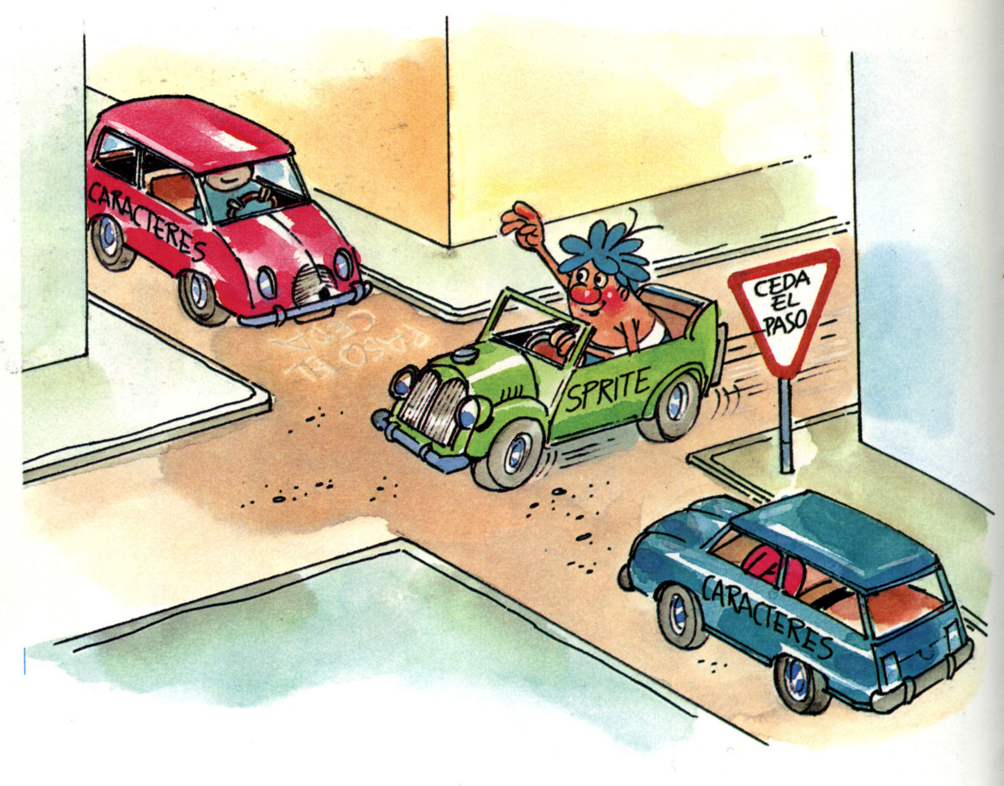
Pues bien, sobre esta estructura, ya conocida, se abren ocho nuevos planos de impresión, todos superiores al del texto en pantalla, y dotados entre ellos de un código de prioridad: los planos correspondientes a los sprites gráficos.

De esta forma, cualquier sprite es de mayor prioridad que los caracteres impresos en la pantalla, por ello, al desplazarse por la pantalla, oculta parcialmente lo ya impreso. Sin embargo, a través de los puntos no activados de la matriz del sprite (color transparente), es posible ver aquellas porciones de carácter que toman el color de la «tinta».

Asimismo, es posible alterar la prioridad existente entre los sprites y lo impreso en la pantalla, de forma



En la pantalla existen tres planos superpuestos: el borde o marco, el fondo y el primer plano.



Cualquier sprite es de mayor prioridad que los caracteres impuestos en la pantalla.

que algunos se deslicen por debajo o por encima de los objetos contenidos en la pantalla, a nuestra voluntad, consiguiéndose de esta forma un efecto tridimensional.

La prioridad asignada a cada sprite frente a los demás es fija, y depende del número del mismo. Así, el sprite 0 está dotado de la más alta prioridad y el 7 de la más baja. Si, por ejemplo, concurren en una misma zona los sprites 3 y 6, es evidente que el primero ocultará, al menos parcialmente, al segundo.

Por otro lado, debemos hacer una matización a la hora de las superposiciones de sprites. Si suponemos que el sprite de mayor prioridad tiene puntos transparentes (no dotados de color), siempre podremos ver a través de ellos parcialmente al que esté oculto bajo él.

Del mismo modo, si el sprite oculto contiene espacios transparentes, que coincidan con los del de más alta prioridad, podremos ver parcialmente lo impreso en el fondo de la pantalla.

Es posible alterar la prioridad entre los sprites y lo impreso a voluntad, consiguiéndose un efecto tridimensional.

A este conjunto de efectos visuales, los cuales tienen por objeto el facilitar una visión tridimensional, se les denomina efecto «ventana».

Al encender el ordenador la prioridad asignada para los objetos contenidos en la pantalla, es mínima en referencia a la de los sprites; de esta forma, cualquiera de ellos se superpone a lo impreso en la misma.

Nosotros podemos alterar este estado de cosas, tanto de forma general como particular, puesto que cada uno posee un bit de control dentro del registro de la posición de memoria 53275. Cada bit a 0 implica la prioridad del sprite frente a los datos, mientras que los bits a 1 suponen lo contrario.

DETECCIÓN DE COLISIONES ENTRE SPRITES

Independientemente del tamaño de los sprites, el VIC II gestiona de forma automática el control de colisiones entre todos ellos, o entre cualquiera de ellos y los datos contenidos en la pantalla.

Este control, tiene en cuenta únicamente las posiciones visibles del sprite, es decir, los puntos más exteriores dotados de color (no transparentes).

Las colisiones entre sprites se controlan a través del registro 53278 del VIC II, el cual tiene incluso en cuenta las habidas fuera de los límites visibles de la pantalla. Dentro de este registro, cada sprite dispone de un bit de control que, de estar a 1, implica que el sprite se encuentra envuelto en una colisión.

Bastará, por lo tanto, con efectuar la lectura del contenido de esta posición, para tener cumplida información del tipo de colisión habido, simple o múltiple. Sin embargo, debemos tener en cuenta que la lectura de este registro por **PEEK,** o su equivalente en código máquina, restaura automáticamente su valor.

Al encender el ordenador, la prioridad asignada a los objetos contenidos en la pantalla es mínima con respecto a la de los sprites.



Por ello, puede ser conveniente almacenar el resultado de su lectura en otra zona de la memoria RAM o incluirlo en una variable BASIC, para permitir un análisis más detenido.

DETECCIÓN DE COLISIONES ENTRE SPRITES Y DATOS

De forma similar a lo comentado para las colisiones entre sprites, existe un registro encargado de controlar las posibles colisiones de sprites con los objetos contenidos en la pantalla, la 53279 del VIC II.

Dentro de ella, cada sprite dispone de un bit que, de estar a 1, implica la existencia de una colisión. También, de forma análoga a lo que sucede con la posición 53278, ésta es de reposición automática al efectuarse su lectura, por lo que resulta recomendable la misma norma comentada para el apartado anterior.

En el modo multicolor, la comprobación es similar, aunque debemos tener especial cuidado puesto que el VIC II considera los datos 01 como transparentes.

UN PROGRAMA DE JUEGO QUE MANEJA SPRITES

Para sembrar un poco de práctica en medio de tanta teoría, podemos entrar en el comentario de un sencillo programa de ejemplo, en el que se muestran algunas características descritas en éste y el capítulo anterior (al final del artículo).

Este juego tiene por objeto, recoger el mayor número posible de tesoros del fondo del mar, para remolcarlos hasta el barco nodriza. Valiéndonos para ello de un batiscafo, el cual debe evitar ser atrapado por los pulpos guardianes del tesoro.

Para aumentar la dificultad, al completar cada pantalla, se incrementa el número de cefalópodos, establecido en uno de comienzo. Es posible, sin embargo, aumentar este número, modificando la asignación de valor a la variable **F,** en la línea 100.

El movimiento hacia arriba y abajo se controla con las teclas 1 y 2, y hacia izquierda y derecha con las del 9 y el 0.

En la línea 20 se borra la pantalla y se establece el valor inicial para la variable **SC$,** la cual tiene por objeto facilitar el posicionamiento en cualquier línea de la pantalla, por medio de la sentencia **LEFT$.** En la línea 30 se completa el contenido de **SC$,** al tiempo que se define una cadena de espacios duros **(SHIFT SPACE),** necesarios en el transcurso del programa.

La línea 40 contiene la dimensión de la matriz **A()** donde se contienen los datos de los **sprites,** al tiempo que se inicializa la variable **V** con el valor de comienzo del chip de video, y la línea 50 lee una cadena de 25 posiciones, desde las líneas **DATA** del final del texto BASIC.

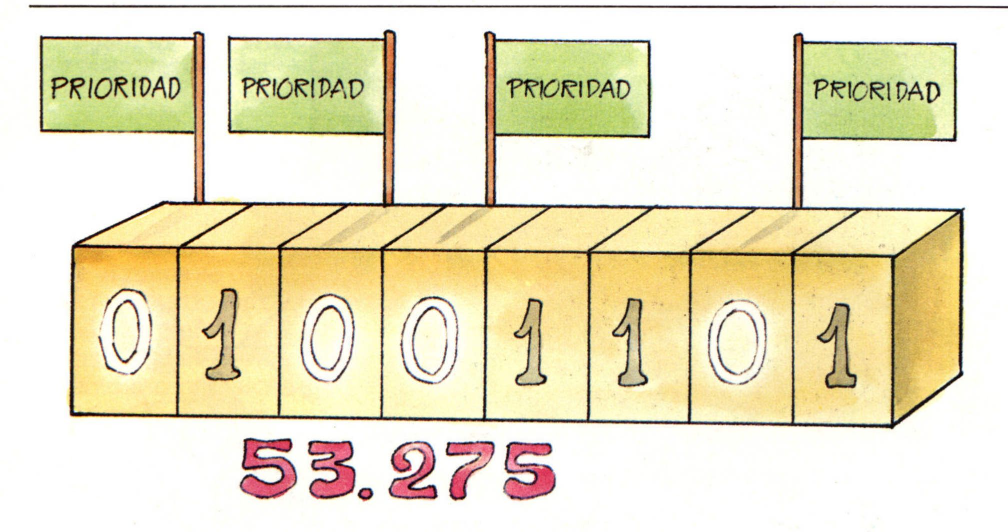
La línea 60 establece el comienzo del bloque de memoria que contendrá las configuraciones de puntos de los sprites, la 70 el color, y la 80 la carga física de la configuración contenida en las sentencias **DATA** en el banco correspondiente.

La línea 100, inicializa los valores de algunas variables, como el número de pulpos activos **F** o el contador de puntos obtenidos **H.**

Las líneas 110 y 120 tienen por objeto colocar el fondo marino con sus tesoros, así como configurar el color de la parte superior de la pantalla.



La prioridad asignada a cada sprite respecto a los demás es fija; depende del número del mismo.



Cada bit de la posición 53275 implica la prioridad de un sprite frente al de datos. Para el control de colisiones el VIC II tiene sólo en cuenta las posiciones visibles del sprite.

Escribe también el último byte de la pantalla, ángulo inferior derecho, por medio de la sentencia **POKE,** al no poder utilizar **PRINT,** pues provocaría el scroll de la misma.

En las líneas 130 a 160, se inicializa el valor de la abscisas de los sprites, a la vez que se imprimen los marcadores de la primera línea de pantalla. Los espacios contenidos en estas líneas son también **SHIFT SPACE.** Por último, la línea 160 imprime los valores numéricos.

Las líneas 170 y 180 se encargan de posicionar inicialmente el barco de la superficie en el extremo izquierdo, desplazándolo lentamente hasta hacerle alcanzar el centro de la pantalla.

Las líneas 190 y 200, se encargan de actualizar la variable que contiene el número de sprites activados **G,** incrementando el número de pulpos en uno por cada nueva pantalla; además, inicializan el valor de arranque del batiscafo en la superficie, pasando estos valores de ordenada y abscisa a su lugar en la matriz **A().**

El bucle contenido en las 210 a 230, establece, a su vez, la posición de partida y desplazamiento, de cada uno de los pulpos activados, de forma aleatoria.

Las líneas 250 a 300, se encargan de efectuar el movimiento de cada uno de los sprites activos, teniendo en cuenta su rebote contra el contorno de la pantalla, evitando que éstos puedan salirse de sus límites.

Las líneas 310 a 380, exploran el teclado para decodificar el movimiento del batiscafo, impidiendo los desplazamientos más allá del contorno de la pantalla.

El bloque de líneas 390 a 450 cierran el bucle principal del programa, controlando la ocurrencia de colisiones.

Las líneas 460 a 510, canalizan los mensajes de fin de programa, efectuando la pregunta de si se desea jugar una nueva partida.

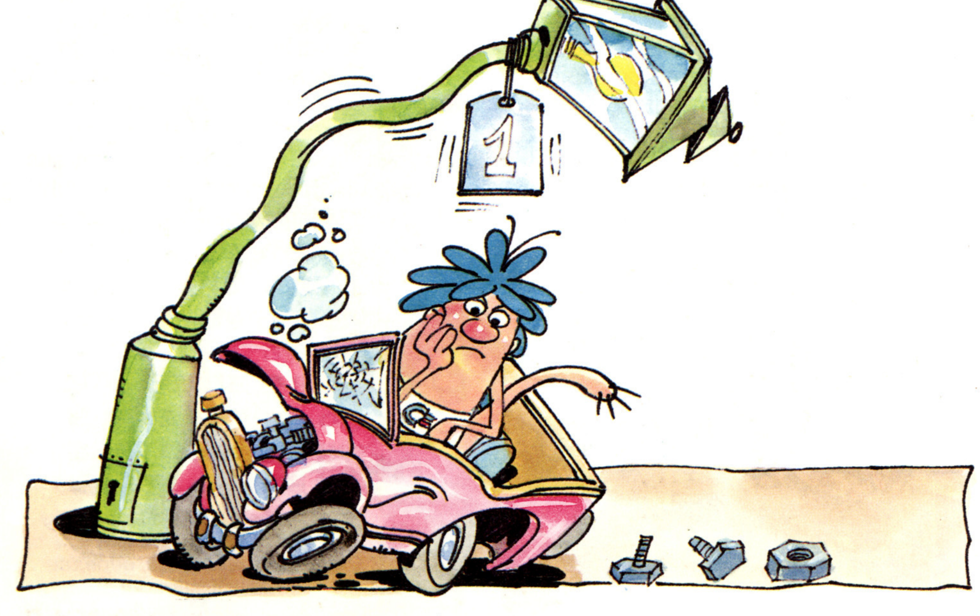
El bloque de líneas 520 a 700, controlan la recogida de los tesoros, añadiendo éstos a la configuración del sprite de batiscafo, y actualizan el marcador de puntos.

Las líneas 710 a 810 contienen el área de subrutinas del programa, retardo, impresión de marcadores con ceros a la izquierda, etc.

Por último, las líneas 820 a 920 contienen la configuración de los sprites.



Cuando cargamos un programa, la pantalla se desconecta, tomando la parte útil de ésta el mismo color que el asignado para el borde o marco.



CONEXIÓN/DESCONEXIÓN DE LA PANTALLA

Si somos un poco observadores, nos habremos dado cuenta de que cuando se ejecuta una operación de carga desde la unidad de cinta la pantalla se desconecta, tomando la

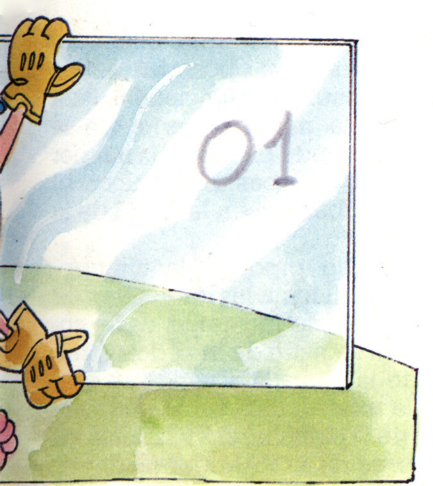
Es conveniente almacenar el resultado de los **PEEK** *a la condición de los sprites en otra zona de RAM o en una variable BASIC, para permitir su análisis más detenido.*

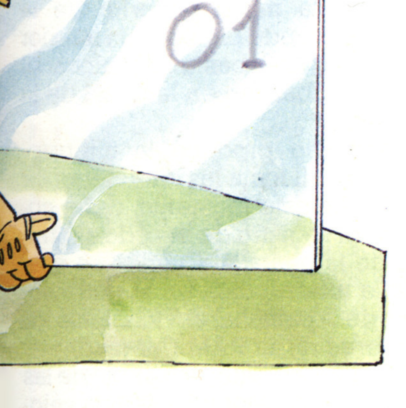
En el modo multicolor debemos tener especial cuidado, porque VIC II considera los datos 01 como transparentes.

parte útil de ésta el mismo color que el asignado para el borde o marco.

En este estado el contenido de la pantalla no se pierde, sino que tan sólo no se nos muestra, con objeto de aliviar transitoriamente el trabajo del microprocesador, para que éste pueda ejecutar otras tareas más rápidamente.

Para comprender mejor la razón de todo esto, debemos pensar que la pantalla del COMMODORE ocupa una zona considerable de la memoria





RAM, con la que el microprocesador debe operar muy activamente. Por ello, el hecho de desocuparse, al menos temporalmente, de esta tarea, redunda en beneficio de la velocidad de ejecución del microprocesador.

Por estas razones, puede ser una buena técnica desactivar la pantalla, mientras se realizan tareas complejas de cálculo, sin requerir la representación de resultados.

La conexión/desconexión de la pantalla se controla a través del bit 4 del registro de control del VIC II 53265. Cuando este bit está a 1, la pantalla es visible normalmente; en caso contrario su representación queda inhibida. Las sentencias BASIC que desconectan y vuelven a conectar la pantalla son las siguientes:

```
POKE 53265,PEEK(53265) AND 239
POKE 53265,PEEK(53265) OR 16
```

REGISTRO DE BARRIDO DE LA PANTALLA

El registro de barrido está localizado en la posición 53266 del VIC II. Con su lectura, se obtienen los 8 bits menos significativos (peso bajo), de la posición actual de barrido de la pantalla (refresco).

El bit restante, el más significativo, se encuentra en la posición 53265. El conocimiento del contenido de este registro de barrido, puede ayudar a efectuar modificaciones en la pantalla sin generar turbulencias, o sea, cuando el barrido no está presente en la zona visible de la misma.

Otra utilidad de este registro es la de almacenar un valor sobre él, teniendo en cuenta por supuesto el bit más significativo. En este caso, el valor almacenado en esta posición se



Cada sprite dispone de un byte de control que, de estar a 1, implica que el sprite se encuentra envuelto en una colisión.

Es una buena técnica desactivar la pantalla mientras se realizan tareas complejas de cálculo.



compara con la posición de barrido, activándose en caso afirmativo un bit en el registro de interrupción del VIC II, en la posición 53273.

REGISTRO DE INTERRUPCIÓN

En la posición 53273 se encuentra la información relativa a cualquier tipo de interrupción. El bit 2 controla la colisión entre sprites, colocándose a 1 cuando ésta se produce, activándose el bit 7 IRQ al producirse una interrupción.

El significado de los tres bits menos significativos es el siguiente: el 0 IRST se activa cuando coincide el barrido con el valor almacenado, el 1 IMDC por las colisiones entre sprites y datos, el 2 IMMC por colisiones entre sprites y el 3 ILP está reservado para uso del lápiz óptico, cuando se dispone de este periférico.

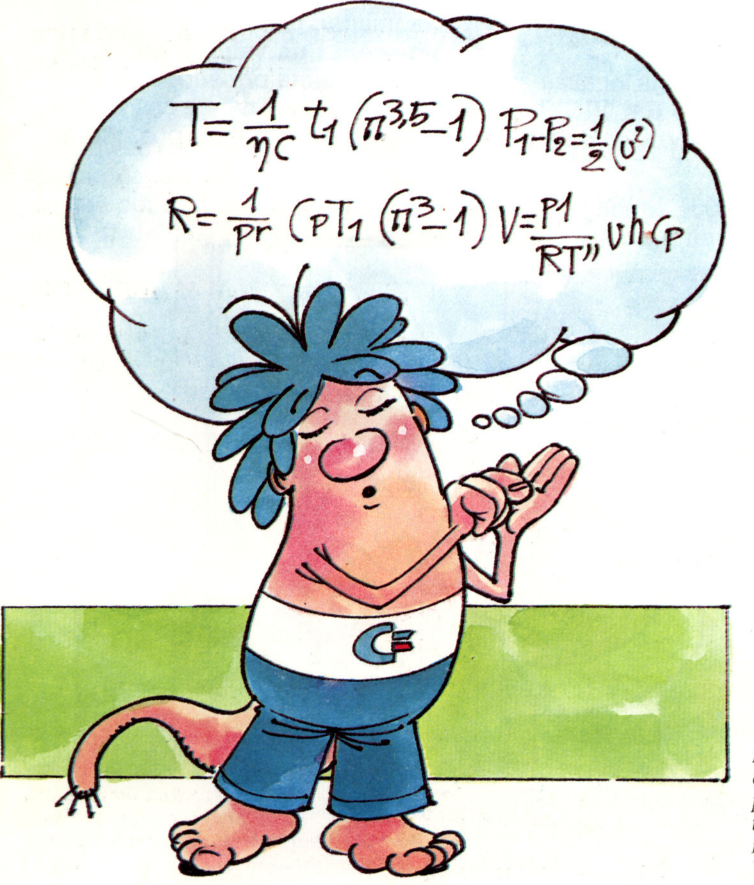
El registro de interrupciones se aloja en la posición de memoria 53274, y está dotado del mismo formato que el registro de estado de interrupción. Ninguna interrupción puede acontecer, a menos que el correspondiente bit del registro de interrupciones se encuentre activado.

Gracias a este registro, puede dividirse la pantalla en zonas, texto o bit-map, por ejemplo; manejar más de 8 sprites, etc.

Manejando las interrupciones adecuadamente, son posibles estas combinaciones de funciones. Para conseguir la mitad superior de la pantalla en modo bit-map y la inferior en el modo texto, basta con colocar el registro de barrido en la mitad de la pantalla.

Cuando se produce la interrupción, el VIC II toma los caracteres de la ROM. En este momento, debe colocarse el registro de barrido al principio de la pantalla. Cuando se presenta la interrupción en la parte superior de la pantalla, el VIC II toma los caracteres de la RAM (bit-map).

En cualquier caso, el manejo de las interrupciones requiere de una velocidad muy grande en la ejecución del programa de control, por ello, no es posible hacerlo desde el BASIC, debiendo recurrir a un programa en código máquina. Sin embargo, lo espectacular de los resultados, bien vale un pequeño esfuerzo adicional de programación.



La conexión y desconexión de la pantalla se controla a través del bit 4 del registro de control del VIC II 53265.

COMBINACIONES IDÓNEAS DE COLOR

El COMMODORE está dotado de una configuración de caracteres, que facilita su legibilidad en pantalla. Para ello, éstos van dotados de verticales dobles (dos puntos de ancho), frente a horizontales simples.

Con esta técnica, los caracteres son más fácilmente legibles, que si estuvieran formados por líneas simples de un solo punto, lo que elimina

Una de las posibilidades gráficas de que disponemos a la hora de programar, es la definición del scroll uniforme.



problemas sobre todo con receptores de T.V.

En cualquier caso, la legibilidad de la representación en pantalla, depende en gran medida de los colores que especifiquemos para el fondo y primer plano. Podríamos decir que existen tres tipos de combinaciones, las ideales, las buenas y las no tan buenas.

Esta calificación de contrastes, entre colores de caracteres y fondo de pantalla, se encuentra reflejada en el siguiente cuadro:

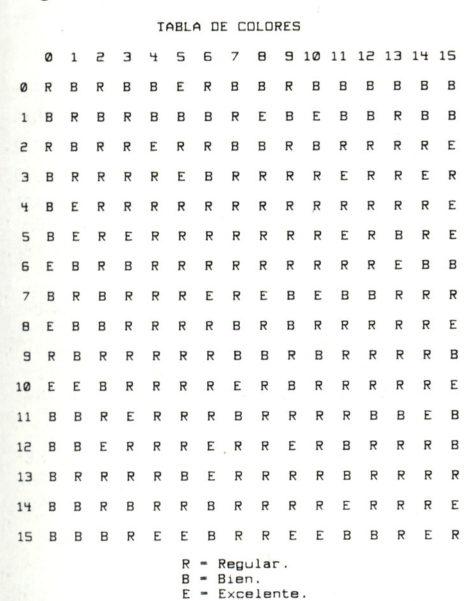


TABLA DE COLORES

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	R	B	R	B	B	E	R	B	B	R	B	B	B	B	B	B
1	B	R	B	R	B	B	B	R	E	B	E	B	B	R	B	B
2	R	B	R	R	E	R	R	B	B	R	B	R	R	R	R	E
3	B	R	R	R	R	E	B	R	R	R	R	E	R	R	E	R
4	B	E	R	R	R	R	R	R	R	R	R	R	R	R	R	E
5	B	E	R	E	R	R	R	R	R	R	R	E	R	B	R	E
6	E	B	R	B	R	R	R	R	R	R	R	R	R	E	B	B
7	B	R	B	R	R	R	E	R	E	B	E	B	B	R	R	R
8	E	B	B	R	R	R	R	B	R	B	R	R	R	R	R	E
9	R	B	R	R	R	R	R	B	B	R	B	R	R	R	R	B
10	E	E	B	R	R	R	R	E	R	B	R	R	R	R	R	E
11	B	B	R	E	R	R	R	B	R	R	R	R	B	B	E	B
12	B	B	E	R	R	R	E	R	R	E	R	B	R	R	R	B
13	B	R	R	R	R	B	E	R	R	R	R	B	R	R	R	R
14	B	B	R	B	R	R	B	R	R	R	R	E	R	R	R	E
15	B	B	B	R	E	E	B	R	R	E	E	B	B	R	E	R

R = Regular.
B = Bien.
E = Excelente.

SCROLL UNIFORME DE LA PANTALLA

Otra de las posibilidades gráficas de que disponemos a la hora de programar, es la de definir el scroll uniforme, o desplazamiento punto a punto, de toda la pantalla en las cuatro direcciones.

De este modo, podemos introducir información en la pantalla de forma progresiva, mientras que la información ya almacenada se va perdiendo por el otro extremo. Esta posibilidad es soportada por los registros de scroll del VIC II; que además posee un modo de 38 columnas y otro de 24 líneas, como soporte para ejecutar esta función.

Para lograr un scroll uniforme de la pantalla es preciso seguir los siguientes pasos. Primero reducir la pantalla, después colocar el registro de scroll al máximo o mínimo, según convenga. A continuación, situaremos los datos en la parte no visible de la pantalla, para incrementar o disminuir el registro de scroll, hasta que alcance su valor máximo o mínimo.

Llegados a este punto, es preciso echar mano de una simple rutina escrita en código máquina, encargada de desplazar todo el contenido de la pantalla, un caracter en la dirección del scroll, para continuar, a partir de este punto, otra vez con la serie de operaciones comentadas.

Para colocar el ordenador en el modo de 38 columnas, debe ponerse a 0 el bit 3 de la posición 53270:

```
POKE 53270,PEEK(53270) AND 247
```

Para volver al modo de 40 columnas, debemos colocar a 1 el correspondiente bit:

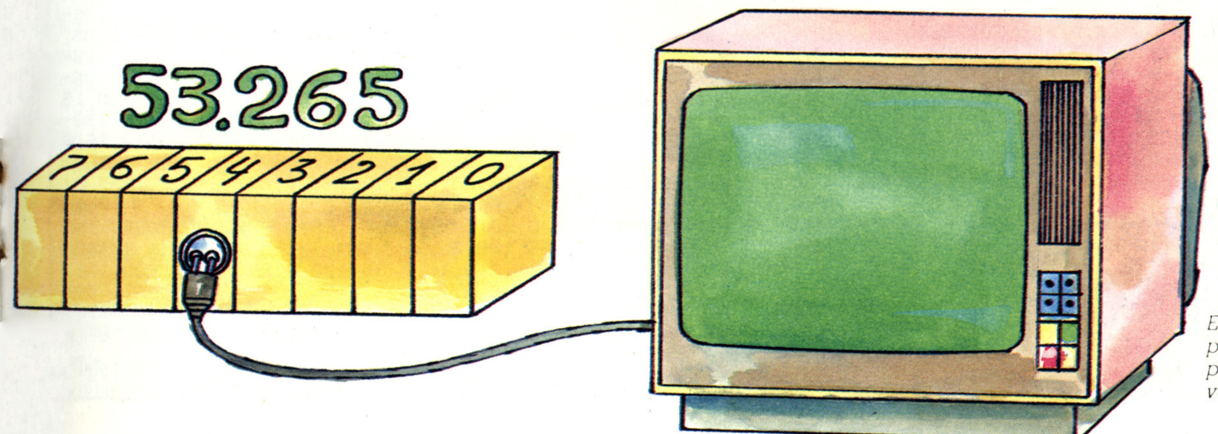
```
POKE 53270,PEEK(53270) OR 8
```

Para pasar al modo de 24 líneas, es preciso desactivar el bit 3 de la posición de memoria 53265:

```
POKE 53265,PEEK(53265) AND 247
```

Para volver al modo normal de 25 líneas, deberemos activar el bit correspondiente:

```
POKE 53265,PEEK(53265) OR 8
```



El Commodore está dotado, para mayor legibilidad en la pantalla, de caracteres con verticales dobles.

Para el scroll horizontal, debemos actuar sobre los bits del registro 53270, en la forma:

POKE 53270,(PEEK(53270) AND 248)+X

Donde **X** representa la posición de la pantalla, de 0 a 7.

Para el scroll vertical, debemos actuar sobre los bits del registro 53265, en la forma:

POKE 53265,(PEEK(53265) AND 248)+Y

Donde **Y** representa la posición de la pantalla, de 0 a 7.

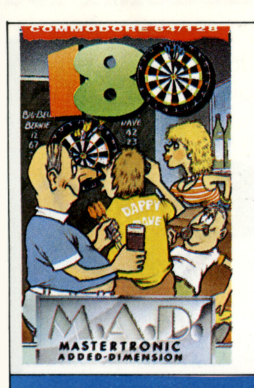
Para provocar un scroll del texto desde abajo de la pantalla, deben colocarse los tres bits menos significativos de la posición 53265, de 0 a 7, a continuación colocar los datos en la línea cubierta inferior de la pantalla, y repetir el proceso mecánicamente.

De forma análoga, para producir un scroll de izquierda a derecha, debemos hacer variar los tres bits menos significativos de la posición 53270, de 0 a 7, escribir la columna oculta de datos, y repetir el proceso mecánicamente.

LISTADO

```
10 REM - REMOLCADOR (C) J.M.LOPEZ MARTINEZ          -149-
20 PRINT"{CLR}":SC$="{HOM}"                         -110-
30 FORI=0TO23:SC$=SC$+"{ABJ}":NEXT:FORI=0TO39:SP$=SP$
+" ":NEXT                                           -127-
40 DIMA(7,3):V=53248                                -089-
50 FORI=0TO24:READJ:B$=B$+CHR$(J):NEXT              -078-
60 J=2040:POKEJ,13:POKEJ+1,14:FORI=2TO7:POKEI+J,15:NE
XT                                                  -155-
70 POKEV+40,9:POKEV+44,13:POKEV+29,2                -149-
80 FORI=0TO191:READJ:POKE832+I,J:NEXT               -108-
90 POKEV+32,15:POKEV+34,14:POKEV+35,8:POKEV+17,91
                                                    -176-
100 E=113:F=1:H=0                                   -165-
110 A$="{YEL}{RON} {ROF}$${RON}[2 ESP]{ROF}$${RON}[2
ESP]{ROF}$${RON}[2 ESP]{ROF}$${RON}[2 ESP]{ROF}$${RON
}[2 ESP]{ROF}$${RON}[2 ESP]{ROF}$${RON}[2 ESP]{ROF}$$
{RON}[2 ESP]{ROF}$${RON}[2 ESP]{ROF}$$"             -206-
120 L=10:F=F+1:N=N+1:PRINT"{CLR}{BLK}"SP$SP$SP$:PRINT
SC$A$;:POKE2023,160                                 -002-
130 POKEV+16,0                                      -039-
140 PRINT"{HOM}{BLK}¦P¦¦U¦¦N¦¦T¦¦O¦¦S¦"CHR$(186)" "TA
B(15)"¦R¦¦E¦¦C¦¦O¦¦R¦¦D¦"CHR$(186)" ";              -136-
150 PRINTTAB(30)"¦I¦¦N¦¦T¦¦E¦¦N¦¦T¦¦O¦"CHR$(186)" "
                                                    -136-
160 J=7:X=H:GOSUB750:J=22:X=D:GOSUB750              -112-
170 POKE1063,E:POKEV+30,0                           -203-
180 POKEV+21,2:POKEV+3,53:C=.5:FORJ=0TO160:POKEV+2,J:
GOSUB730:NEXT                                       -105-
190 G=3:IFF<8THENFORI=2TOF:G=G+2^I:NEXT             -045-
200 A(0,0)=172:A(0,1)=76:POKEV,172:POKEV+1,76
                                                    -227-
210 FORI=2TOF:A(I,0)=RND(1)*294+25:A(I,1)=RND(1)*146+
75                                                  -027-
220 A(I,2)=4*(INT(RND(1)*3)-1):A(I,3)=4*(INT(RND(1)*3
)-1):IFA(I,2)=0THEN220                              -037-
230 NEXT                                            -244-
240 J=7:X=H:GOSUB750                                -225-
250 FORI=2TOF:A(I,0)=A(I,0)+A(I,2):A(I,1)=A(I,1)+A(I,
3)                                                  -212-
260 IFA(I,0)<25ORA(I,0)>319THENA(I,2)=A(I,2)*-1
                                                    -058-
270 IFA(I,1)<75ORA(I,1)>221THENA(I,3)=A(I,3)*-1
                                                    -060-
280 A=INT(A(I,0)/256):B=A(I,0)-A*256:IFATHENPOKEV+16,
PEEK(V+16)OR2^I:GOTO300                             -246-
290 POKEV+16,PEEK(V+16)AND(255-2^I)                 -034-
300 POKEV+2*I,B:POKEV+2*I+1,A(I,1):NEXT:POKEV+21,G
                                                    -254-
310 GETX$:IFX$=""THEN360                            -158-
320 IFX$="1"THENA(0,2)=0:A(0,3)=-4:GOTO360          -051-
330 IFX$="2"THENA(0,2)=0:A(0,3)=4:GOTO360           -008-
340 IFX$="9"THENA(0,3)=0:A(0,2)=-4:GOTO360          -061-
350 IFX$="0"THENA(0,3)=0:A(0,2)=4                   -252-
360 X=A(0,0)+A(0,2):Y=A(0,1)+A(0,3)                 -242-
370 IFX<21ORX>323THEN420                            -212-
380 IFY<74ORY>228THEN420                            -227-
390 A=INT(A(0,0)/256):B=A(0,0)-A*256:IFATHENPOKEV+16,
PEEK(V+16)OR1:GOTO410                               -032-
400 POKEV+16,PEEK(V+16)AND254                       -195-
410 POKEV,B:POKEV+1,Y:A(0,0)=X:A(0,1)=Y             -036-
420 X=PEEK(V+30):IFX-INT(X/2)*2THEN520              -010-
430 IFA(0,1)=228THEN580                             -010-
440 IFA(0,1)=76THEN630                              -216-
450 GOTO250                                         -137-
460 PRINTLEFT$(SC$,12)TAB(4)"{WHT}DESEA INTENTARLO DE
 NUEVO (S/N)?"                                      -045-
470 IFH>DTHEND=H:J=23:X=D:GOSUB750                  -155-
480 GETX$:IFX$<>"S"ANDX$<>"N"THEN480                -148-
490 IFX$="S"THEN100                                 -092-
500 PRINTLEFT$(SC$,12)TAB(4)"[7 ESP]GRACIAS POR JUGAR
![7 ESP]"                                           -080-
510 END                                             -141-
520 C=.5:FORJ=0TO4:POKEV+21,PEEK(V+21)AND254:GOSUB730
                                                    -222-
530 POKEV+21,PEEK(V+21)OR1:NEXT:POKEV+21,PEEK(V+21)AN
D254:GOSUB800:E=E+1                                 -140-
540 IFMTHENGOSUB780                                 -227-
550 IFL=0THEN660                                    -205-
560 IFE=116THEN460                                  -045-
570 GOTO170                                         -141-
580 IFMTHEN250                                      -095-
590 A=(A(0,0)-28)/32:IFA<>INT(A)THEN250             -157-
600 B=A*6+5:IFMID$(A$,B,1)<>"$"THEN250              -038-
610 A$=LEFT$(A$,B-1)+"[2 ESP]"+MID$(A$,B+2):PRINTSC$A
$;:GOSUB800                                         -246-
620 H=H+(A+1)*N:FORI=1TO25:POKE870+I,ASC(MID$(B$,I,1)
):NEXT:M=1:L=L-1:GOTO240                            -051-
630 IFM=0THEN250                                    -200-
640 IFA(0,0)<157ORA(0,0)>186THEN250                 -162-
650 H=H+10*N                                        -140-
660 GOSUB780:GOSUB800:IFLTHEN240                    -007-
670 POKEV+21,2:C=.5:FORJ=161TO344                   -201-
680 A=INT(J/256):B=J-A*256:IFATHENPOKEV+16,PEEK(V+16)
OR2:GOTO700                                         -125-
690 POKEV+16,PEEK(V+16)AND253                       -205-
700 POKEV+2,B:POKEV+3,53:GOSUB730:NEXT:POKEV+2,0:POKE
V+16,0:GOTO110                                      -175-
710 POKE54273,A:POKE54272,B:GOSUB730                -136-
720 RETURN                                          -153-
730 FORI=0TO99*C:NEXT                               -082-
740 RETURN                                          -155-
750 J=J+1024:X$=RIGHT$(STR$(X),LEN(STR$(X))-1):X$=RIG
HT$("[4 0]"+X$,5)                                   -157-
760 FORI=1TO5:POKEI+J,ASC(MID$(X$,I,1))+64:NEXT
                                                    -089-
770 RETURN                                          -158-
780 FORI=1TO25:POKE870+I,0:NEXT:M=0                 -172-
790 RETURN                                          -160-
800 A(0,2)=0:A(0,3)=0                               -013-
810 RETURN                                          -153-
820 DATA3,129,192,1,0,128,3,193,224,4,2,0,3,129,192,0
,64,32,7,131,192,1,0,128                            -251-
830 DATA0                                           -005-
840 DATA0,255,0,3,255,192,7,255,224,15,255,240,28,231
,56,28,231,56,31,255,248                            -036-
850 DATA15,255,240,7,255,224,3,189,192,6,0,96,6,0,96,
3,0,192,0,0,0,0,0,0,0,0                             -028-
860 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0               -108-
870 DATA0,0,0,0,16,0,0,48,0,0,16,0,0,16,0,0,56,0,0,16
,0,4,16,64,4,16,64,14,56                            -198-
880 DATA224,4,16,64,4,16,64,4,16,64,31,125,240,4,16,6
4,228,16,64,245,211,64                              -186-
890 DATA255,255,255,237,182,220,127,255,248,63,255,24
0,0                                                 -041-
900 DATA0,30,0,0,63,128,0,255,128,1,255,192,3,255,192
,3,255,192,3,255,128,3                              -169-
910 DATA147,136,1,219,16,0,254,92,0,84,180,0,239,136,
7,255,192,12,254,96,9,233                           -090-
920 DATA48,26,101,144,18,180,152,34,84,200,35,82,118,
31,41,137,1,64,146,0                                -077-

READY.                                              -173-
```



180

Hoy puede parecer un día normal, pero en Quintic Village, pequeña población del norte de Escocia, es una jornada muy especial. Todo estriba en la inusual apuesta que se ha cruzado entre los personajes más singulares del pueblecito: Big Fat Gordon y Billy Boy Jim. El premio para el ganador será de 100 pintas de la mejor cerveza en el tugurio de Sam, pagadas por supuesto por el perdedor. La competición se centrará en un match de dardos, al mejor de tres juegos, y se celebrará, como no, en la taberna de Sam. ¿Quién será capaz de perdérselo?



La hora señalada ha llegado, y la gente se agolpa en las escasas sillas de que dispone la taberna. Miles de apuestas se entrecruzan y los murmullos se recrudecen bajo las voces de uno que asegura la victoria de Big Fat u otro que no duda que el vencedor será Billy Boy. Y de pronto: el silencio. La puerta se ha abierto y en el umbral aparece la oronda figura de Big Fat, sobre el que caen una salva de vítores y hurras. Con un gesto de su mano hace callar a la masa, y con gran aplomo se dirige a la barra: Sam, ponme una de tus pintas.

De pronto, la estilizada figura de Billy Boy irrumpe en el establecimiento y el silencio se hace ahora más poderoso. Los dos contendientes se miran y Big Fat es el primero en abrir la boca: Esas 100 pintas me van a proporcionar una buena borrachera, gracias muchacho. A lo que éste responde: Primero habrás de ganarlas, tras lo cual ambos se dirigen hacia el escenario del match secundados por un estallido de júbilo general.

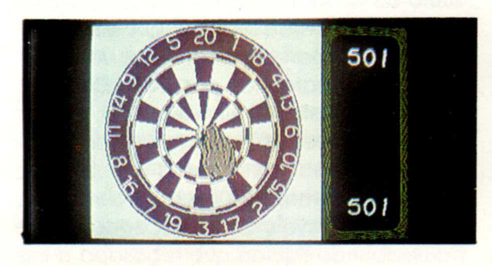
La competición empezará en 501 puntos, a los cuales se les irán restando los conseguidos en las dianas, tirando tres dardos cada vez. La máxima puntuación en 180 (tres triples 20) y cada juego ha de acabar en Ø justo y con el último dardo en un doble.

Tras una hora, las cosas están al rojo vivo: un juego ha caido de cada bando y en el último, el definitivo, Big Fat se dispone a tirar, para restar ya de 25. El primer dardo, ante el estupor general cae al suelo, el segundo acierta en el 9 simple, y... ¡Magnífico, el tercero ha caido sobre el doble 8!, Big Fat ha vencido! Pero la voz de Billy Boy surge entre el vocerío y sentencia: Aún debo tirar yo. Y ante la sorpresa de todo el pueblo, Billy, que resta de 151, realiza un triple 17, un triple 20, y su último dardo vuela certero y se clava en el doble 17 ¡Ha empatado! Big Fat mira asombrado a la diana, y Billy exclama: Tranquilo, Fatty, nos cogeremos la borrachera juntos. Dicen que la fiesta se prolongó hasta el día siguiente y miles de historias se cuentan sobre esa noche, porque... ¿hay algo más emocionante que un match de dardos?



VEREDICTO FINAL

GRAFICOS	****
SONIDO	***
ORIGINALIDAD	***
DIFICULTAD	****
INTÉRES	****





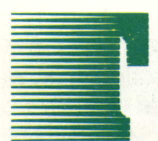
FICHA TECNICA

NOMBRE	180
PRECIO	699 PTAS.
SOPORTE	CINTA
TIPO	DEPORTES
MODELO	C-64, C-128
OBSERVACIONES	NINGUNA



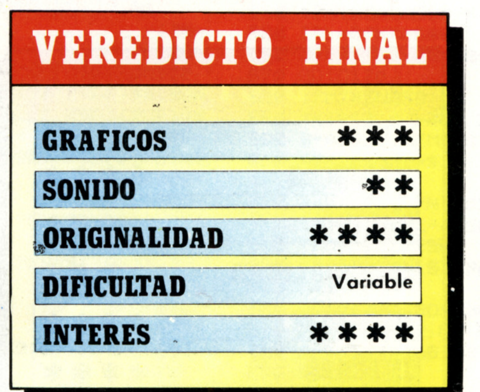
Mad Nurse

La organización es lo más importante: es la frase más usual que surge de la mofletuda boca del director del hospital. Su estertorea voz va dejando caer estas palabras a través de los pasillos, en la más inesperada esquina, escondido, acechante tras las mamparas de las salas de enfermos irrecuperables. Oyéndole hablar parecería que ni el Pentágono pudiera competir en perfección organizativa con este sanatorio, lo cual casi es cierto, a no ser por ese habitáculo, ¡sí!, ese antro de descontrol y desorden escondido en el laberinto de la primera planta... la guardería.



Todo empezó con un insulso y escueto anuncio en un periódico de segunda clase: «Se precisa personal para guardería en hospital, seriedad y presencia de ánimo, plazas limitadas». Recordando ahora este episodio de mi vida, desde las cuatro paredes de mi habitáculo sito en el prestigioso Manicomio para Deshauciados de Mentes Desquiciadas (M.D.M.D), me parecen muy lejanas las razones por las que me decidí a aceptar el empleo. El caso es que me planté en las oficinas de admisión y ante mi sorpresa me aceptaron sin poner trabas. Sólo ahora entiendo aquella frase de la empleada de guardia comentando el tema con su compañera: ¡Parecía tan joven y feliz!

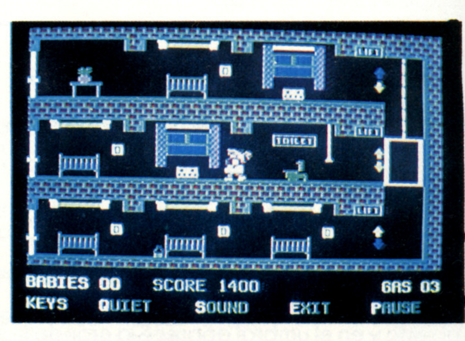
En ese momento no lo entendí y me dirigí a mi trabajo con la mayor diligencia. El letrero de la puerta estaba muy limpito y el silencio era total; una sensación de paz y bienestar invadió mi cuerpo, y, con gran decisión, hice lo que nunca debí haber hecho, el principio de mi caida en el negro pozo de la desesperación: ¡abrí la puerta!

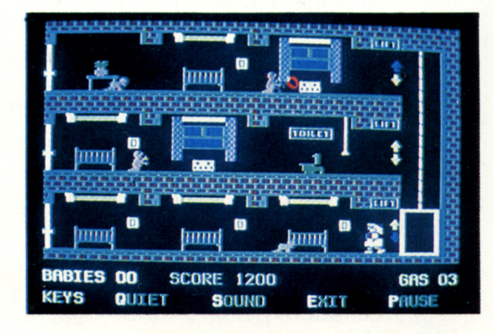


VEREDICTO FINAL

GRAFICOS	***
SONIDO	**
ORIGINALIDAD	****
DIFICULTAD	Variable
INTERES	****

En ese preciso instante observé que el suelo de la habitación estaba plagado de decenas de formas blancas, que con sus vivaces ojitos iban arrastrándose, unos reptando, otros gateando por el blanco suelo: eran bebés; éstos, al oír mi entrada, giraron sus lindas cabecitas hacia mí y se convirtieron en unas salvajes bestias que saltaron sobre mis piernas, me llenaron de babas, mordiscos e involuntarios escapes de ese nauseabundo líquido que rebosaba a través de sus pañales.





FICHA TECNICA

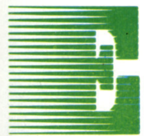
NOMBRE	MAD NURSE
PRECIO	699 PTAS.
SOPORTE	CINTA
TIPO	ACCION
MODELO	C-64, C-128
OBSERVACIONES	NINGUNA

Acto seguido, escaparon en todas direcciones, dejándome en un extraño estado de aturdimiento y humedad. Se dispersaron por todos los rincones, yendo hacia los enchufes y botes con productos tóxicos de la sala, y por cada uno que metía en la cuna, saltaban otros ocho al suelo. Y lo peor de todo es que siguen allí, otros diferentes, pero seguramente más malignos, y no creo que haya nadie capaz de detenerlos.



Flash Gordon

El mundo se halla bajo un peligro mortal que amenaza con destruirlo y convertirlo en una nube de polvo estelar. Quedan 24 horas para que el final se produzca, y nadie sabe si podrá evitarse. La esperanza está depositada en un hombre que ha dado muestras otras veces de un valor más allá de lo humano, si alguien es capaz de lograr lo imposible, ese es... ¡Flash Gordon!



El cruel Ming, deseoso de hacerse con el control de todo el Universo, y ante la denodada resistencia que la Tierra impone a su irrefrenable avance, se ha decidido por la vía rápida. Dirigiendo sus potentes misiles hacia el centro de la Tierra, logrará desintegrarla por completo en el terriblemente corto plazo de 24 horas.

La Federación ha enviado a Flash a buscarle: si Ming vive, la Tierra morirá, y viceversa, ya que éste lleva consigo, como medida de seguridad, el control remoto que activa los misiles, y si Flash quiere arrancárselo y desactivarlos, necesitará previamente haber acabado con él, cosa nada fácil, por cierto.

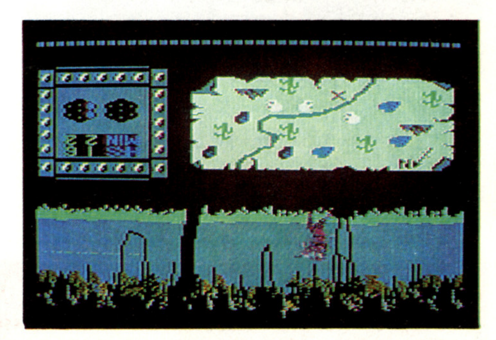
Empezaremos nuestra misión en la jungla del mundo de Ming, ya que nuestra nave se ha estropeado, cayendo entre la maleza, afortunadamente sin perjuicio alguno para nosotros. Ante todo, deberemos encontrar la cueva del príncipe Barín, el único que quizás pueda ayudarnos. Para llegar hasta él deberemos superar la multitud de dificultades que interpondrá la selva, con miles de ojos escrutándonos entre la espesura. Si nuestra habilidad, reflejos y valentía es tan legendaria como pregonan, llegaremos a las cavernas de Barín. Este es un personaje muy desconfiado y terco, y tendremos que ganarnos su respeto con el objeto de que nos deje un jet y nos enseñe la forma de llegar hasta Ming.

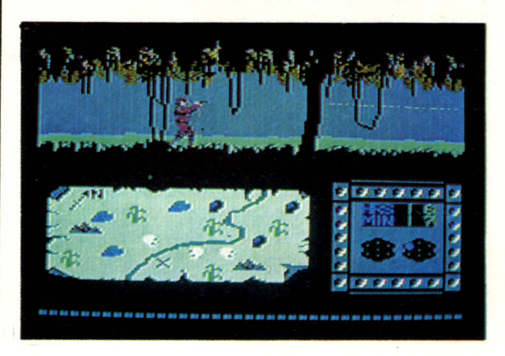
Estas dos etapas, parecen difíciles, pero no serán nada comparadas con el encuentro final: nuestras fuerzas contra las de Ming y sus guardianes. La lucha tendrá lugar en el aire, y sus jets nos parecerán pequeños, pero rápidamente podremos comprobar que los partidarios de Ming no pertenecen al club de fans Flash Gordon. Esta batalla será a muerte, y por supuesto sólo habrá un único vencedor, y sólo un perdedor, o Ming o la Tierra. Pero mejor será no entretenerse demasiado, el tiempo pasa... ¡y sólo nos restan 24 horas!

VEREDICTO FINAL

GRAFICOS	****
SONIDO	***
ORIGINALIDAD	****
DIFICULTAD	*****
INTERES	****

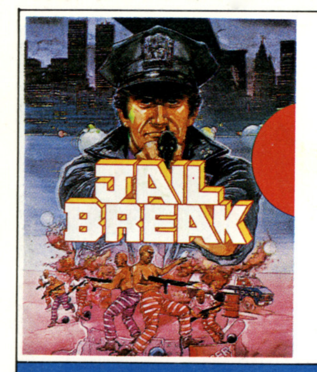






FICHA TECNICA

NOMBRE	FLASH GORDON
PRECIO	699 PTAS.
SOPORTE	CINTA
TIPO	ACCION
MODELO	C-64, C-128
OBSERVACIONES	NINGUNA



Jail Break

La penitenciaría de Freeworld, conocida por sus increíbles medidas de seguridad, de la que no ha conseguido escapar con vida preso alguno, se halla en un estado de gran efervescencia debido a la prohibición de jugar a las cartas en los espacios de recreo. Los presos han sentido violados sus derechos más personales, y se está preparando un motín... ¿tendrá éxito?



Los mejores cerebros en cuestión de fugas se han reunido en secreto en una asamblea que ha tenido lugar en los servicios de la planta 4, la de los estafadores, pensando en las diferentes posibilidades de éxito de una fuga generalizada. Después de desechar como inviables la del túnel al exterior, por ser muy arriesgada y difícil de llevar a cabo con la totalidad de reclusos, y la del ataque masivo contra los guardias, debido al gran número de bajas que podría acarrear, se ha decidido tomar como rehén a una persona importante, para lograr salir al exterior sin lucha ni meses de demora.

Y la elección ha recaido en la persona del alcaide de la prisión, enérgico y respetado por sus subordinados. El hecho tuvo lugar en la comida de cumpleaños de éste, por ser el único día que todos los presos se hallaban juntos en un mismo lugar y acompañados del alcaide, y tal como estaba planeado, el éxito fue rotundo y la totalidad de reclusos consiguió escapar.

El estado de alerta en la ciudad limítrofe con la penitenciaría es total, ya que los evadidos campan por sus respetos por sus calles, sin nadie que se atreva a hacerles frente, ya que además han conseguido tomar otros muchos rehenes entre la población civil. No se sabe cómo, pero han logrado hacerse con armas y representan un increíble peligro.

Así que la lucha se traslada a la calle, y como buenos policías que somos, deberemos acabar con los fugados, que dispararán contra nosotros a la mínima posibilidad que tengan. Tendremos asimismo el deber de rescatar a los rehenes que tiene en su poder; por cada uno que arranquemos de sus garras, se nos proporcionará un arma suplementaria. Es una misión más difícil de lo que parece, ya que son muchos, y se hallan apostados tras los coches, escondidos en las alcantarillas, en los tejados de las casas o empuñando una pistola en plena calle ¿lograremos salir indemnes de... Jail Break?

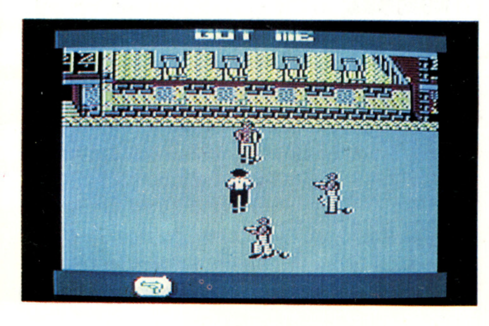
VEREDICTO FINAL

GRAFICOS	***
SONIDO	**
ORIGINALIDAD	****
DIFICULTAD	****
INTERES	***

FICHA TECNICA

NOMBRE	JAIL BREAK
PRECIO	2.200 PTAS.
SOPORTE	CINTA
TIPO	ACCION
MODELO	C-64, C-128
OBSERVACIONES	NINGUNA









Microrhythm

El concierto acaba de comenzar, en un principio, el público se halla expectante ante la posible perfección que puedan desarrollar los músicos en su interpretación. Sólidas baladas acompañan a perfectos y vivos rock & rolls que sumergen al público en la música hasta hacerle perder el sentido del tiempo. Pero sobre todo, en este concierto hay un instrumento que subyuga a la vociferante masa: con sus increíbles ritmos, la batería les hace moverse a todos bajo un mismo son. Acercarse al sonido de una batería no es fácil, pero Microrhythm es tan real... que muchos lo dudarían.



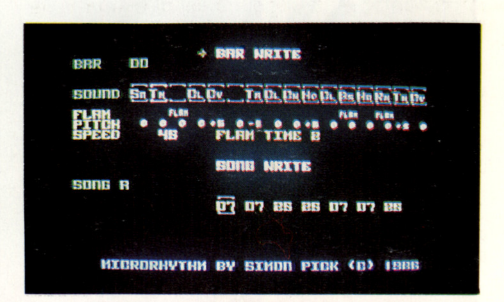
Este programa consta de tres modos: el modo de composición, en el cual podremos componer los ritmos de las canciones con la velocidad, tono y duración que prefiramos; el modo de compás, con el cual podremos ajustar nuestras ideas a unos compases continuados y el de tiempo real, para lograr por fin escuchar nuestra creación.

Los instrumentos que pueden ser utilizados en este programa son: el bombo, tambor, tres tipos de timbales, platillos altos-cerrados, platillos altos-abiertos, charleston, gong, tambor de borde, cencerro, palmas y pandereta. Podremos comprobar que con ellos es posible realizar multitud de composiciones diferentes y de muy distintas tonalidades.

Además, gozaremos de otras muchas ventajas como pueden ser: el Flam, que consiste en la repetición o eco de un sonido, el cual puede ser subido o bajado según nuestras preferencias. También se permite subir o bajar el tono, así como reducir o aumentar la duración de un compás, para que quede a nuestro gusto. Incluso, con el objeto de no despistarnos, podremos escuchar un metrónomo que nos orientará acerca de la velocidad que queremos imponerle a nuestra composición e impedirá que nos adelantemos al ritmo o retrasemos éste.

Por último, y como punto final, para aquellos enamorados de sus creaciones, tendremos la ocasión de grabar nuestras propias canciones, para escucharlas, si así lo deseamos, posteriormente. Sin duda, Microrhythm nos ofrece una buena cantidad de oportunidades de ejercitarnos en el sentido del ritmo y contrastar unos sonido con otros muy diferentes, por ello, un solo consejo: ¡pruébalo, no te arrepentirás!







VEREDICTO FINAL

GRAFICOS	*
SONIDO	*****
ORIGINALIDAD	*****
DIFICULTAD	***
INTERES	*****

FICHA TECNICA

NOMBRE	MICRORHYTHM
PRECIO	699 PTAS.
SOPORTE	CINTA
TIPO	MUSICA
MODELO	C-64, C-128
OBSERVACIONES	NINGUNA

Datassette

Cómo obtener el máximo provecho del cassette de Commodore

Comenzamos este mes una serie de artículos, para explicar cómo se guarda la información en los dipositivos periféricos de almacenamiento masivo: el cassette y el disco de los ordenadores Commodore.

En primer lugar, trataremos el periférico de almacenamiento por excelencia: la cinta. Para completar los conocimientos expuestos en teoría sobre él mismo, incluiremos un programa en código máquina de gran utilidad, que nos permitirá descubrir: el nombre, dirección de inicio, dirección de final, longitud total, carga relativa o absoluta y tipo de fichero grabado en cinta.

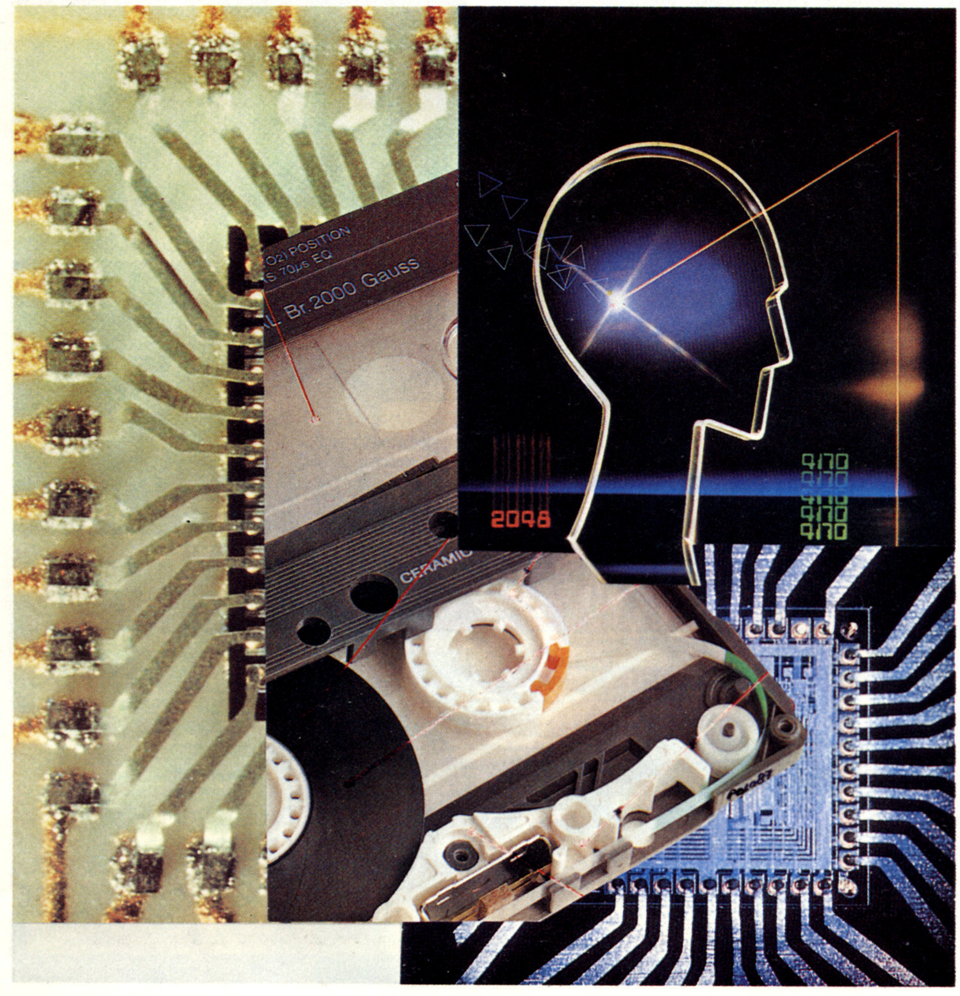
Como sabemos, la información en el interior del ordenador se almacena en forma de ceros y unos (bits), que se agrupan en una unidad superior que consta de ocho bits denominada byte u octeto.

El problema básico consiste en estudiar la forma de almacenar esta información volátil (ya que se encuentra en la RAM), de una forma semipermanente (con capacidad de borrado y modificación), en algún soporte que permita un acceso sencillo y seguro a dichos datos.

En el caso que nos ocupa, se trata de transformar estos ceros y unos (existencia o ausencia de tensión) en una serie de impulsos eléctricos, modificados por una bobina en impulsos magnéticos capaces de alterar la orientación magnética de unos materiales llamados ferromagnéticos (fig. 1).

Los materiales empleados en las cintas son de este tipo, y tienen la particularidad de «recordar» la información sobre la polaridad del campo magnético a la que han sido sometidos, permaneciendo esta información hasta que son expuestos a otro campo magnético de características análogas que lo anule.

Como es lógico esto ocurre durante la escritura; el proceso de lectura es justamente al contrario. Los cambios en el sentido de la polaridad en la cinta, excitan a la bobina captadora que los transforma en impulsos eléctricos, estos amplificados de forma conveniente son inteligibles para el ordenador (fig. 2).



Requisitos Hardware y Software

Pero todo este proceso no es tan sencillo, para que esta señal se pueda grabar en cinta, es necesario que cumpla una serie de requisitos que vienen impuestos tanto por la física como por el hardware y el software incluídos en la memoria ROM del ordenador, tales como: nivel de ruido, amplitud de la señal, codificación, duración de los impulsos, etc.

Comenzaremos por los requisitos físicos. Aquí encontraremos los motivos por los cuales no es posible la utilización de una grabadora normal de audio para nuestro cometido, lo cual no viene impuesto por la forma del conector. Aunque básicamente la grabadora/reproductora digital funciona bajo los mismos principios que las de audio, existen una serie de matices a tener en cuenta.

— Nivel de ruido. Si el ruido que produce nuestro reproductor/grabador es superior a los mínimos de señal, es posible que estos ruidos sean entendidos como señales digitales, por lo que introduciríamos errores al quedar la señal original enmascarada, debemos saber que es posible que un fallo en un único bit, puede llevar al traste un programa entero.

— Amplitud de la señal. Si la señal es demasiado débil, es posible que el sistema la rechace considerándola como ruido de fondo, en el caso del C-64 las señales óptimas oscilan entre los 0V. y los 5V.

— La frecuencia. Es posible que el sistema no sea capaz de reconocer por motivos físicos (banda de frecuencias) una señal que no tenga la frecuencia necesaria.

El problema de la señal/ruido se resuelve con un circuito específico, denominado disparador Schmitt, este circuito, normalmente se encuentra en la placa bajo la forma de un integrado, y es la diferencia básica entre un circuito de audio analógico y otro digital.

El disparador, en esencia, transforma una señal variable en una de flancos rápidos como es el caso de una onda cuadrada. Las tensiones de alimentación nos condicionan la tensión máxima y mínima, para que el disparador identifique una señal como cero o como uno (fig. 3).

Con frecuencia, se utilizan circuitos disparadores inversores que funcionan de la siguiente manera: ante una señal de entrada que reconocen como uno, la sacan como cero, por lo que si queremos obtener la misma señal de entrada a la salida de una serie de circuitos inversores, tendremos que conectar uno a continuación de otro en un número par de ellos. He aquí el primer motivo por el que no podemos conectar dos grabadoras/reproductoras Commodore de forma directa para intercambio directo de información, sin incluir una interface entre las mismas (fig. 4).

La segunda razón estriba en que los ordenadores Commodore envían la información de forma invertida a como la reciben (fig. 5), por lo que si queremos conectar dos grabadoras Commodore directamente, tendremos que hacerlo a través de un inversor (fig. 5), y si la queremos conectar a una grabadora normal de audio, podremos hacerlo a través de dos inversores, que adaptaran la señal oportunamente (fig. 6).

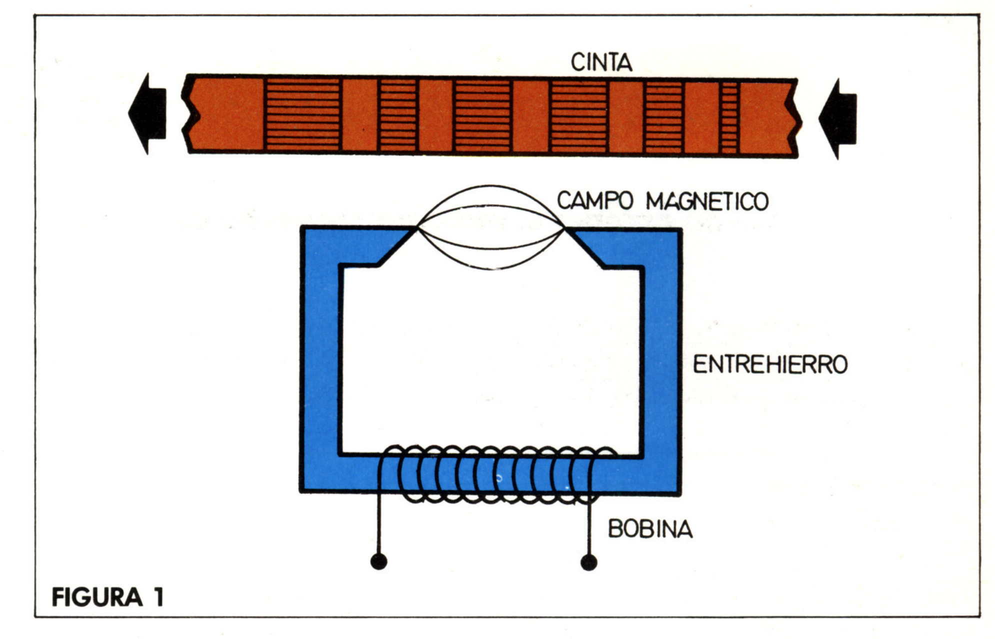


FIGURA 1

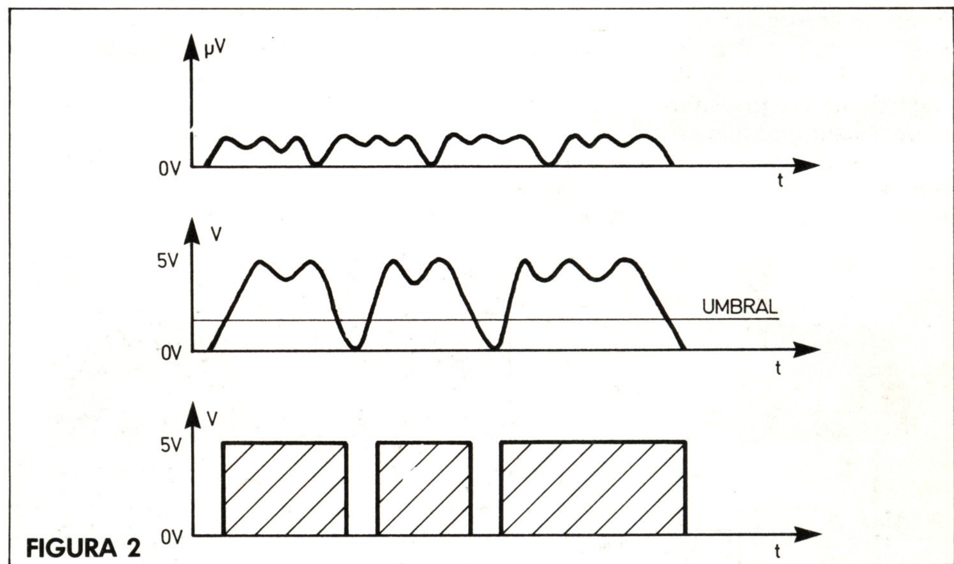


FIGURA 2

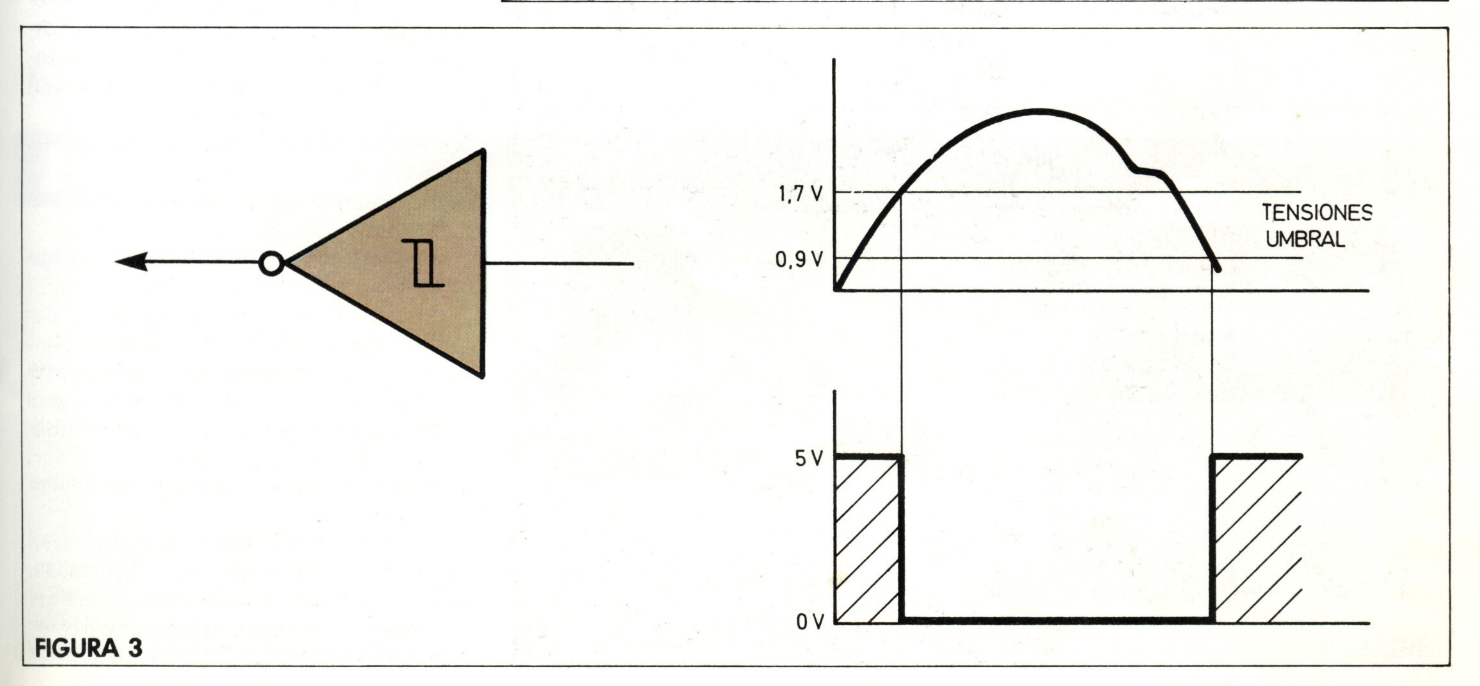


FIGURA 3

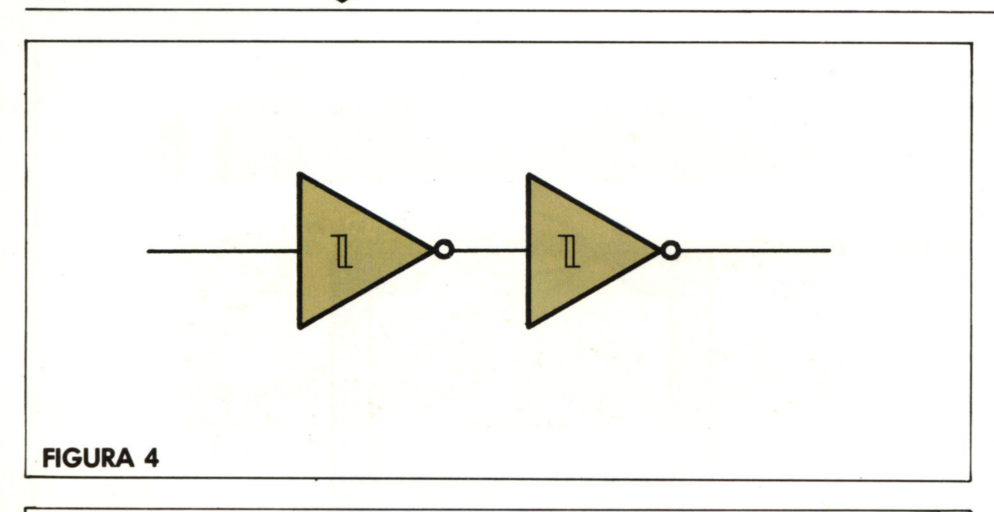


FIGURA 4

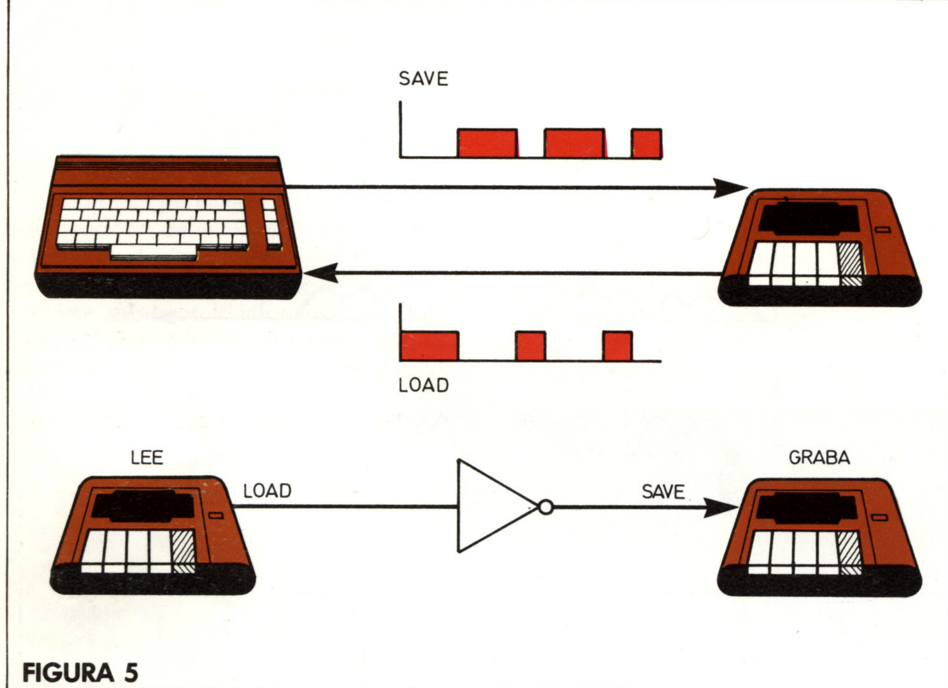


FIGURA 5

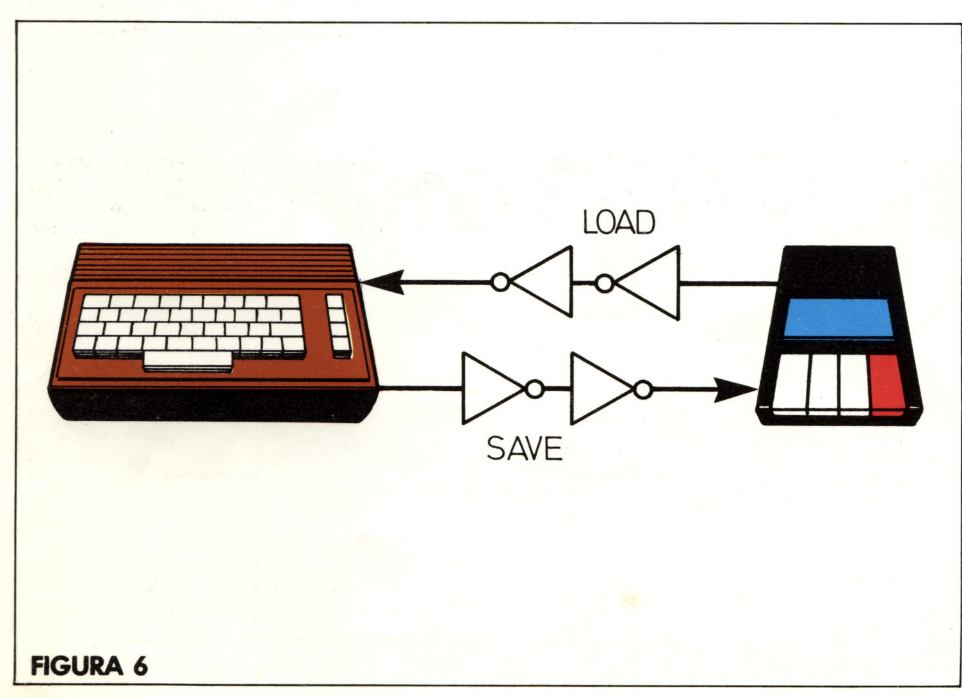


FIGURA 6

La grabadora/reproductora de los Commodore dispone de un circuito integrado TTL 74LS14 que consta de seis inversores conectados según la figura 7.

Todo esto se complica bastante si pensamos en los requisitos que nos impone el software y el hardware, como es el caso de las demás líneas de interface que incorpora el datassette aparte de las de LOAD y SAVE; a saber: las dos de alimentación de 0V (masa) y de 5V, la de SENSE que detecta la pulsación de una tecla del datassette (avisa al ordenador que puede iniciar la emisión de datos hacia el periférico), y la de CASS que pone en marcha el motor bajo el control del ordenador.

Otro de los requisitos del hardware/software, es el formato de los datos. Normalmente éstos no se envían de una forma desordenada, sino que siguen unas pautas muy severas. Veamos cómo se manda un byte (8 bits), al ordenador o desde el ordenador.

Lo primero que tenemos que saber es que los Commodore emplean tres tipos de impulsos de distinta duración para codificar la información:

Impulso A de 176 microsegundos.
Impulso B de 256 microsegundos.
Impulso C de 336 microsegundos.

Para codificar un byte, primeramente se envía una serie CCBB que precede a cualquier byte, luego un 1 se representa por una serie BBAA y un 0 por AABB (fig. 8). 0.96 milisegundos por cada carácter.

El buffer

Un elemento importante de la transmisión de datos desde el datassette y hacia él, es el buffer del cassette; una zona de memoria que va desde la posición $033C hasta la $1019 (en el C-64), cuya misión es generar la cabecera de identificación, la cual tiene el siguiente formato:

Byte 0 posición 828 tipo de fichero.
Significado:
01 programa carga relativa (BASIC).
02 bloque de datos.
03 programa carga absoluta (código máquina).
04 cabecera de datos.
05 bloque EOT (fin de cinta).

Byte 1,2 posición 829,830 dirección de inicio LO/HI.

Byte 3,4 posición 831,832 dirección de final LO/HI.

Byte 5 posición 833 en adelante nombre de fichero.

El resto se llena de espacios, hasta el final del buffer. Además, ésta es la mínima unidad de transferencia de información entre el datassette y el ordenador, por lo cual se utilizan estos bytes sobrantes para programas

cargadores con «autorun» escondidos en la cabecera.

Asimismo, el buffer tiene la misión de almacenar los bloques de información que mediante rutinas de Kernal, van pasando al registro acumulador del microprocesador y de allí al datassette o, en el proceso de carga, del datassette al acumulador y de allí al buffer.

Si nos elevamos más en el formato de grabación de datos en el datassette, veremos que los bloques están integrados en unidades mayores. Antes de la cabecera propiamente dicha va una serie de bytes de sincronización (sirven para poner en marcha un contador de impulsos que lleva el control de los bytes que entran o salen de buffer), luego nos encontramos con la cabecera, donde hallamos los datos del fichero, más adelante otro grupo de sincronización y los bloques de datos, y por último, puede haber un bloque final indicador de final de cinta (bloque EOT).

El control de errores en el caso de los ordenadores Commodore es muy efectivo. El primero se produce a nivel de byte, tras el cual hay un bit de paridad. El siguiente se efectúa sobre bloque, en el que se genera un byte de control tras una serie de operaciones lógicas con todos los datos. Por si todo esto fuera poco, toda la información es repetida de nuevo para comparar si lo que se ha transmitido coincide en los dos casos.

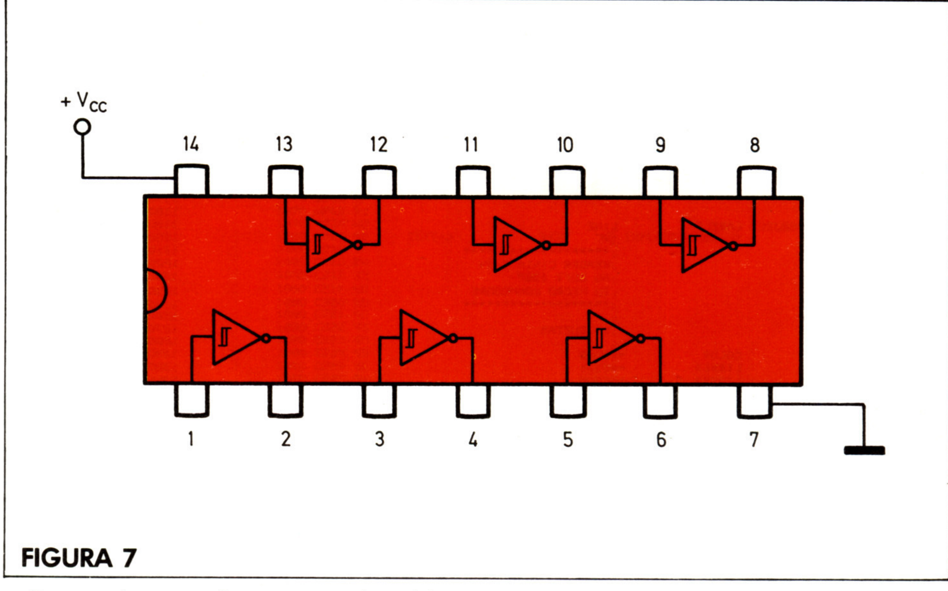


FIGURA 7

Este es el principal motivo por el cual las operaciones con el datassette son tan lentas. Lógicamente, los turbos disponibles, eliminan parte de estas medidas de seguridad, a la par que aumentan la velocidad de transmisión, por lo que es frecuente la aparición de errores en este tipo de operaciones con turbo.

Esperamos que este artículo haya servido de ayuda, para comprender cómo funcionan internamente en las operaciones de la lectura escritura con el datassette los ordenadores Commodore.

Para finalizar, y como ya adelantamos al comienzo de estas páginas, he aquí un programa que pone en práctica todo lo visto sobre el datassette, efectuando un completísimo análisis de las cintas. Adjuntamos igualmente un listado ensamblador del mismo, con el fin de que aquellos lectores entre cuyos conocimientos se cuente el código máquina, puedan proceder a un estudio más exhaustivo de nuestro programa HEADER.

INDICADOR BYTE	1	Ø	Ø	1	Ø	Ø	Ø	1
C C B B	B B A A	A A B B	A A B B	B B A A	A A B B	A A B B	A A B B	B B A A

FIGURA 8

LISTADO

```
10 A=0:PRINT"(CLR)(11 DCH)F.A. ACERO (C) 1987":PRINT
                                                 -237-
20 FORBYT=49152TO49706                           -071-
30 READVA:POKEBYTE,VA:A=A+VA                     -081-
40 NEXT                                          -195-
50 IF A<>48067THENPRINT"(CLR)ERROR EN DATAS":END
                                                 -102-
60 PRINT"*CARGAR PROGRAMA DE CINTA, CUANDO LO[7 ESP]E
NCUENTRE PULSAR RUN/STOP"                        -173-
70 PRINT"*ACTIVAR CON SYS 49152"                 -208-
80 PRINT"*TRAS VER LOS DATOS"                    -115-
90 PRINT"*REBOBINAR Y CARGAR O BUSCAR OTRO"      -020-
100 PRINT"*EL HEADER TAPE SE ENCUENTRA DE 49152 A[4 E
SP]49706"                                        -134-
110 PRINT"*SI EL PROGRAMA NO LO PISA LO MANTENEMOS[3
ESP]EN MEMORIA"                                  -039-
500 DATA 169,0,141,32,208,141,33,208,32,68,229,169,10
5,160,193,32                                     -189-
510 DATA 30,171,24,162,2,160,6,32,10,229,169,139,160,
193,32,30                                        -026-
520 DATA 171,24,162,4,160,0,32,10,229,169,164,160,193
,32,30,171                                       -075-
530 DATA 169,65,160,3,32,30,171,24,162,10,160,0,32,10
,229,169                                         -230-
540 DATA 172,160,193,32,30,171,173,62,3,174,61,3,32,2
05,189,32                                        -037-
550 DATA 94,193,173,61,3,141,52,3,173,62,3,141,53,3,3
2,28                                             -045-
560 DATA 193,24,162,12,160,0,32,10,229,169,180,160,19
3,32,30,171                                      -128-
570 DATA 173,64,3,174,63,3,32,205,189,32,94,193,173,6
3,3,141                                          -212-
580 DATA 52,3,173,64,3,141,53,3,32,28,193,56,173,63,3
,237                                             -056-
590 DATA 61,3,141,52,3,173,64,3,237,62,3,141,53,3,24,
162                                              -245-
600 DATA 14,160,0,32,10,229,169,187,160,193,32,30,171
,173,53,3                                        -030-
610 DATA 174,52,3,32,205,189,32,94,193,32,28,193,24,1
62,16,160                                        -049-
620 DATA 0,32,10,229,169,199,160,193,32,30,171,173,60
,3,201,1                                         -233-
630 DATA 208,10,169,205,160,193,32,30,171,76,27,193,2
01,2,208,10                                      -129-
640 DATA 169,220,160,193,32,30,171,76,27,193,201,3,20
8,10,169,236                                     -192-
650 DATA 160,193,32,30,171,76,27,193,201,4,208,10,169
,254,160,193                                     -194-
660 DATA 32,30,171,76,27,193,201,5,208,10,169,16,160,
194,32,30                                        -037-
670 DATA 171,76,27,193,169,27,160 194,32,30,171,96,17
3,53,3,32                                        -064-
680 DATA 53,193,173,53,3,32,63,193,173,52,3,32,53,193
,173,52                                          -211-
690 DATA 3,32,63,193,96,32,72,193,32,77,193,32,12,225
,96,41                                           -170-
700 DATA 15,32,77,193,32,12,225,96,74,74,74,74,96,201
,10,48                                           -166-
710 DATA 9,56,233,10,24,105,65,76,93,193,24,105,48,96
,169,102                                         -010-
720 DATA 160,193,32,30,171,96,61,36,0,32,32,32,32,70,
46,65                                            -091-
730 DATA 32,65,67,69,82,79,32,32,84,85,32,77,73,67,82
,79                                              -053-
740 DATA 32,67,79,77,77,79,68,79,82,69,0,72,69,65,68,
69                                               -030-
750 DATA 82,32,67,73,78,84,65,32,32,49,50,45,48,49,45
,49                                              -042-
760 DATA 57,56,55,0,78,79,77,66,82,69,58,0,73,78,73,6
7                                                -219-
770 DATA 73,79,58,0,70,73,78,65,76,58,0,68,73,70,69,8
2                                                -208-
780 DATA 69,78,67,73,65,58,0,84,73,80,79,58,0,66,65,8
3                                                -215-
790 DATA 73,67,44,82,69,76,65,84,73,86,79,0,66,76,79,
81                                               -020-
800 DATA 85,69,32,68,69,32,68,65,84,79,83,0,77,65,81,
85                                               -008-
810 DATA 73,78,65,44,32,65,66,83,79,76,85,84,79,0,67,
65                                               -008-
820 DATA 66,69,67,69,82,65,32,68,69,32,68,65,84,79,83
,0                                               -011-
830 DATA 66,76,79,81,85,69,32,69,79,84,0,78,79,32,73,
77                                               -019-
840 DATA 80,76,69,77,69,78,84,65,68,79,0         -035-
```

```
-----------------------------------------------------------------------
A LISTING, PAGE #1
-----------------------------------------------------------------------

PASS 1

PASS 2

LINE# LOC. OBJECT LABELS           LINE

   10 0000                         ;******************
   20 0000                         ;HEADER CINTA C-64
   30 0000                         ;AO SOFT 1987
   40 0000                         ;TU MICRO COMMODORE
   50 0000                         ;******************
   60 0000                         ;
   70 0000                         ;ETIQUETAS
   80 0000                         ;
   90 0000             COLOR       -$D020
  100 0000             LIBRE3      -$0334
  110 0000             LIBRE4      -$0335
  120 0000             EDITNU      -$BDCD
  130 0000             EDITCA      -$AB1E
  140 0000             BORRA       -$E544
  150 0000             SETCUR      -$E50A
  160 0000             BUFFER      -$033C
  170 0000             CHROUT      -$E10C
  180 0000                         ;
  190 0000                         ;PROGRAMA
  200 0000                         ;
  210 C000                         .ORG    $C000
  220 C000 A900                    LDA #$00
  230 C002 8D20D0                  STA COLOR
  240 C005 8D21D0                  STA COLOR+1
  250 C008 2044E5                  JSR BORRA
  260 C00B A969                    LDA #TEXT1&255
  270 C00D A0C1                    LDY #TEXT1/256
  280 C00F 201EAB                  JSR EDITCA
  290 C012 18                      CLC
  300 C013 A202                    LDX #$02
  310 C015 A006                    LDY #$06
  320 C017 200AE5                  JSR SETCUR
  330 C01A A98B                    LDA #TEXT2&255
  340 C01C A0C1                    LDY #TEXT2/256
  350 C01E 201EAB                  JSR EDITCA
  360 C021 18                      CLC
  370 C022 A204                    LDX #$04
  380 C024 A000                    LDY #$00
  390 C026 200AE5                  JSR SETCUR
  400 C029 A9A4                    LDA #TEXT3&255
  410 C02B A0C1                    LDY #TEXT3/256
  420 C02D 201EAB                  JSR EDITCA
  430 C030 A941                    LDA #$41
  440 C032 A003                    LDY #$03
  450 C034 201EAB                  JSR EDITCA
  460 C037 18                      CLC
  470 C038 A20A                    LDX #$0A
  480 C03A A000                    LDY #$00
  490 C03C 200AE5                  JSR SETCUR
  500 C03F A9AC                    LDA #TEXT4&255
  510 C041 A0C1                    LDY #TEXT4/256
  520 C043 201EAB                  JSR EDITCA
  530 C046 AD3E03                  LDA BUFFER+2
  540 C049 AE3D03                  LDX BUFFER+1

-----------------------------------------------------------------------
A LISTING, PAGE #2
-----------------------------------------------------------------------

LINE# LOC. OBJECT LABELS           LINE

  550 C04C 20CDBD                  JSR EDITNU
  560 C04F 205EC1                  JSR DOLAR
  570 C052 AD3D03                  LDA BUFFER+1
  580 C055 8D3403                  STA LIBRE3
  590 C058 AD3E03                  LDA BUFFER+2
  600 C05B 8D3503                  STA LIBRE4
  610 C05E 201CC1                  JSR DECODE
  620 C061 18                      CLC
  630 C062 A20C                    LDX #$0C
  640 C064 A000                    LDY #$00
  650 C066 200AE5                  JSR SETCUR
  660 C069 A9B4                    LDA #TEXT5&255
  670 C06B A0C1                    LDY #TEXT5/256
  680 C06D 201EAB                  JSR EDITCA
  690 C070 AD4003                  LDA BUFFER+4
  700 C073 AE3F03                  LDX BUFFER+3
  710 C076 20CDBD                  JSR EDITNU
  720 C079 205EC1                  JSR DOLAR
  730 C07C AD3F03                  LDA BUFFER+3
  740 C07F 8D3403                  STA LIBRE3
  750 C082 AD4003                  LDA BUFFER+4
  760 C085 8D3503                  STA LIBRE4
  770 C088 201CC1                  JSR DECODE
  780 C08B 38                      SEC
  790 C08C AD3F03                  LDA BUFFER+3
  800 C08F ED3D03                  SBC BUFFER+1
  810 C092 8D3403                  STA LIBRE3
  820 C095 AD4003                  LDA BUFFER+4
  830 C098 ED3E03                  SBC BUFFER+2
  840 C09B 8D3503                  STA LIBRE4
  850 C09E 18                      CLC
  860 C09F A20E                    LDX #$0E
  870 C0A1 A000                    LDY #$00
  880 C0A3 200AE5                  JSR SETCUR
  890 C0A6 A9BB                    LDA #TEXT6&255
  900 C0A8 A0C1                    LDY #TEXT6/256
  910 C0AA 201EAB                  JSR EDITCA
  920 C0AD AD3503                  LDA LIBRE4
  930 C0B0 AE3403                  LDX LIBRE3
  940 C0B3 20CDBD                  JSR EDITNU
  950 C0B6 205EC1                  JSR DOLAR
  960 C0B9 201CC1                  JSR DECODE
  970 C0BC 18                      CLC
  980 C0BD A210                    LDX #$10
  990 C0BF A000                    LDY #$00
 1000 C0C1 200AE5                  JSR SETCUR
 1010 C0C4 A9C7                    LDA #TEXT7&255
 1020 C0C6 A0C1                    LDY #TEXT7/256
 1030 C0C8 201EAB                  JSR EDITCA
 1040 C0CB AD3C03                  LDA BUFFER
 1050 C0CE C901                    CMP #$01
 1060 C0D0 D00A                    BNE DOS
 1070 C0D2 A9CD                    LDA #TEXT8&255
 1080 C0D4 A0C1                    LDY #TEXT8/256
 1090 C0D6 201EAB                  JSR EDITCA
 1100 C0D9 4C1BC1                  JMP FIN
 1110 C0DC C902    DOS             CMP #$02
 1120 C0DE D00A                    BNE TRES
 1130 C0E0 A9DC                    LDA #TEXT9&255

-----------------------------------------------------------------------
A LISTING, PAGE #3
-----------------------------------------------------------------------

LINE# LOC. OBJECT LABELS           LINE

 1140 C0E2 A0C1                    LDY #TEXT9/256
 1150 C0E4 201EAB                  JSR EDITCA
 1160 C0E7 4C1BC1                  JMP FIN
 1170 C0EA C903    TRES            CMP #$03
 1180 C0EC D00A                    BNE CUATRO
 1190 C0EE A9EC                    LDA #TEXT10&255
 1200 C0F0 A0C1                    LDY #TEXT10/256
 1210 C0F2 201EAB                  JSR EDITCA
 1220 C0F5 4C1BC1                  JMP FIN
 1230 C0F8 C904    CUATRO          CMP #$04
 1240 C0FA D00A                    BNE CINCO
 1250 C0FC A9FE                    LDA #TEXT11&255
 1260 C0FE A0C1                    LDY #TEXT11/256
 1270 C100 201EAB                  JSR EDITCA
 1280 C103 4C1BC1                  JMP FIN
 1290 C106 C905    CINCO           CMP #$05
 1300 C108 D00A                    BNE ERROR
 1310 C10A A910                    LDA #TEXT12&255
 1320 C10C A0C2                    LDY #TEXT12/256
 1330 C10E 201EAB                  JSR EDITCA
 1340 C111 4C1BC1                  JMP FIN
 1350 C114 A91B    ERROR           LDA #TEXT13&255
 1360 C116 A0C2                    LDY #TEXT13/256
 1370 C118 201EAB                  JSR EDITCA
 1380 C11B 60      FIN             RTS
 1390 C11C AD3503 DECODE           LDA LIBRE4
 1400 C11F 2035C1                  JSR MAIN1
 1410 C122 AD3503                  LDA LIBRE4
 1420 C125 203FC1                  JSR MAIN2
 1430 C128 AD3403                  LDA LIBRE3
 1440 C12B 2035C1                  JSR MAIN1
 1450 C12E AD3403                  LDA LIBRE3
 1460 C131 203FC1                  JSR MAIN2
 1470 C134 60                      RTS
 1480 C135 2048C1 MAIN1            JSR SHIFT
 1490 C138 204DC1                  JSR ASCII
 1500 C13B 200CE1                  JSR CHROUT
 1510 C13E 60                      RTS
 1520 C13F 290F    MAIN2           AND #%00001111
 1530 C141 204DC1                  JSR ASCII
 1540 C144 200CE1                  JSR CHROUT
 1550 C147 60                      RTS
 1560 C148 4A      SHIFT           LSR A
 1570 C149 4A                      LSR A
 1580 C14A 4A                      LSR A
 1590 C14B 4A                      LSR A
 1600 C14C 60                      RTS
 1610 C14D C90A    ASCII           CMP #$0A
 1620 C14F 3009                    BMI CIFRA
 1630 C151 38                      SEC
 1640 C152 E90A                    SBC #$0A
 1650 C154 18                      CLC
 1660 C155 6941                    ADC #$41
 1670 C157 4C5DC1                  JMP SAL
 1680 C15A 18      CIFRA           CLC
 1690 C15B 6930                    ADC #$30
 1700 C15D 60      SAL             RTS
 1710 C15E A966    DOLAR           LDA #TEXTO&255
 1720 C160 A0C1                    LDY #TEXTO/256

-----------------------------------------------------------------------
A LISTING, PAGE #4
-----------------------------------------------------------------------

LINE# LOC. OBJECT LABELS           LINE

 1730 C162 201EAB                  JSR EDITCA
 1740 C165 60                      RTS
 1750 C166 3D      TEXTO           .BYTE   '=$',0
 1750 C167 24
 1750 C168 00
 1760 C169 20      TEXT1           .BYTE   '    F.A ACERO  TU MICRO COMMODORE',0
 1760 C16A 20
 1760 C16B 20
 1760 C16C 20
 1760 C16D 46
 1760 C16E 2E
 1760 C16F 41
 1760 C170 20
 1760 C171 41
 1760 C172 43
 1760 C173 45
 1760 C174 52
 1760 C175 4F
 1760 C176 20
 1760 C177 20
 1760 C178 54
 1760 C179 55
 1760 C17A 20
 1760 C17B 4D
 1760 C17C 49
 1760 C17D 43
 1760 C17E 52
 1760 C17F 4F
 1760 C180 20
 1760 C181 43
 1760 C182 4F
 1760 C183 4D
 1760 C184 4D
 1760 C185 4F
 1760 C186 44
 1760 C187 4F
 1760 C188 52
 1760 C189 45
 1760 C18A 00
 1770 C18B 48      TEXT2           .BYTE   'HEADER CINTA  12-01-1987',0
 1770 C18C 45
 1770 C18D 41
 1770 C18E 44
 1770 C18F 45
 1770 C190 52
 1770 C191 20
 1770 C192 43
 1770 C193 49
 1770 C194 4E
 1770 C195 54
 1770 C196 41
 1770 C197 20
 1770 C198 20
 1770 C199 31
 1770 C19A 32
 1770 C19B 2D
 1770 C19C 30
 1770 C19D 31
 1770 C19E 2D
```

```
------------------------------------------------------------------------
A LISTING, PAGE #5
------------------------------------------------------------------------

LINE# LOC. OBJECT LABELS        LINE

 1770 C19F 31
 1770 C1A0 39
 1770 C1A1 38
 1770 C1A2 37
 1770 C1A3 00
 1780 C1A4 4E     TEXT3         .BYTE   'NOMBRE:',0
 1780 C1A5 4F
 1780 C1A6 4D
 1780 C1A7 42
 1780 C1A8 52
 1780 C1A9 45
 1780 C1AA 3A
 1780 C1AB 00
 1790 C1AC 49     TEXT4         .BYTE   'INICIO:',0
 1790 C1AD 4E
 1790 C1AE 49
 1790 C1AF 43
 1790 C1B0 49
 1790 C1B1 4F
 1790 C1B2 3A
 1790 C1B3 00
 1800 C1B4 46     TEXT5         .BYTE   'FINAL:',0
 1800 C1B5 49
 1800 C1B6 4E
 1800 C1B7 41
 1800 C1B8 4C
 1800 C1B9 3A
 1800 C1BA 00
 1810 C1BB 44     TEXT6         .BYTE   'DIFERENCIA:',0
 1810 C1BC 49
 1810 C1BD 46
 1810 C1BE 45
 1810 C1BF 52
 1810 C1C0 45
 1810 C1C1 4E
 1810 C1C2 43
 1810 C1C3 49
 1810 C1C4 41
 1810 C1C5 3A
 1810 C1C6 00
 1820 C1C7 54     TEXT7         .BYTE   'TIPO:',0
 1820 C1C8 49
 1820 C1C9 50
 1820 C1CA 4F
 1820 C1CB 3A
 1820 C1CC 00
 1830 C1CD 42     TEXT8         .BYTE   'BASIC,RELATIVO',0
 1830 C1CE 41
 1830 C1CF 53
 1830 C1D0 49
 1830 C1D1 43
 1830 C1D2 2C
 1830 C1D3 52
 1830 C1D4 45
 1830 C1D5 4C
 1830 C1D6 41
 1830 C1D7 54
 1830 C1D8 49
 1830 C1D9 56

------------------------------------------------------------------------
A LISTING, PAGE #6
------------------------------------------------------------------------

LINE# LOC. OBJECT LABELS        LINE

 1830 C1DA 4F
 1830 C1DB 00
 1840 C1DC 42     TEXT9         .BYTE   'BLOQUE DE DATOS',0
 1840 C1DD 4C
 1840 C1DE 4F
 1840 C1DF 51
 1840 C1E0 55
 1840 C1E1 45
 1840 C1E2 20
 1840 C1E3 44
 1840 C1E4 45
 1840 C1E5 20
 1840 C1E6 44
 1840 C1E7 41
 1840 C1E8 54
 1840 C1E9 4F
 1840 C1EA 53
 1840 C1EB 00
 1850 C1EC 4D     TEXT10        .BYTE   'MAQUINA, ABSOLUTO',0
 1850 C1ED 41
 1850 C1EE 51
 1850 C1EF 55
 1850 C1F0 49
 1850 C1F1 4E
 1850 C1F2 41
 1850 C1F3 2C
 1850 C1F4 20
 1850 C1F5 41
 1850 C1F6 42
 1850 C1F7 53
 1850 C1F8 4F
 1850 C1F9 4C
 1850 C1FA 55
 1850 C1FB 54
 1850 C1FC 4F
 1850 C1FD 00
 1860 C1FE 43     TEXT11        .BYTE   'CABECERA DE DATOS',0
 1860 C1FF 41
 1860 C200 42
 1860 C201 45
 1860 C202 43
 1860 C203 45
 1860 C204 52
 1860 C205 41
 1860 C206 20
 1860 C207 44
 1860 C208 45
 1860 C209 20
 1860 C20A 44
 1860 C20B 41
 1860 C20C 54
 1860 C20D 4F
 1860 C20E 53
 1860 C20F 00
 1870 C210 42     TEXT12        .BYTE   'BLOQUE EOT',0
 1870 C211 4C
 1870 C212 4F
 1870 C213 51
 1870 C214 55
```

```
------------------------------------------------------------------------
A LISTING, PAGE #7
------------------------------------------------------------------------

LINE# LOC. OBJECT LABELS        LINE

 1870 C215 45
 1870 C216 20
 1870 C217 45
 1870 C218 4F
 1870 C219 54
 1870 C21A 00
 1880 C21B 4E     TEXT13        .BYTE   'NO IMPLEMENTADO',0
 1880 C21C 4F
 1880 C21D 20
 1880 C21E 49
 1880 C21F 4D
 1880 C220 50
 1880 C221 4C
 1880 C222 45
 1880 C223 4D
 1880 C224 45
 1880 C225 4E
 1880 C226 54
 1880 C227 41
 1880 C228 44
 1880 C229 4F
 1880 C22A 00

SUCCESSFUL ASSEMBLY; NO ERRORS.


A LISTING, PAGE #8
========================================================================

CONCORDANCE TABLE IN ALPHABETICAL ORDER:

LABEL        VALUE DEFN. REFERENCES

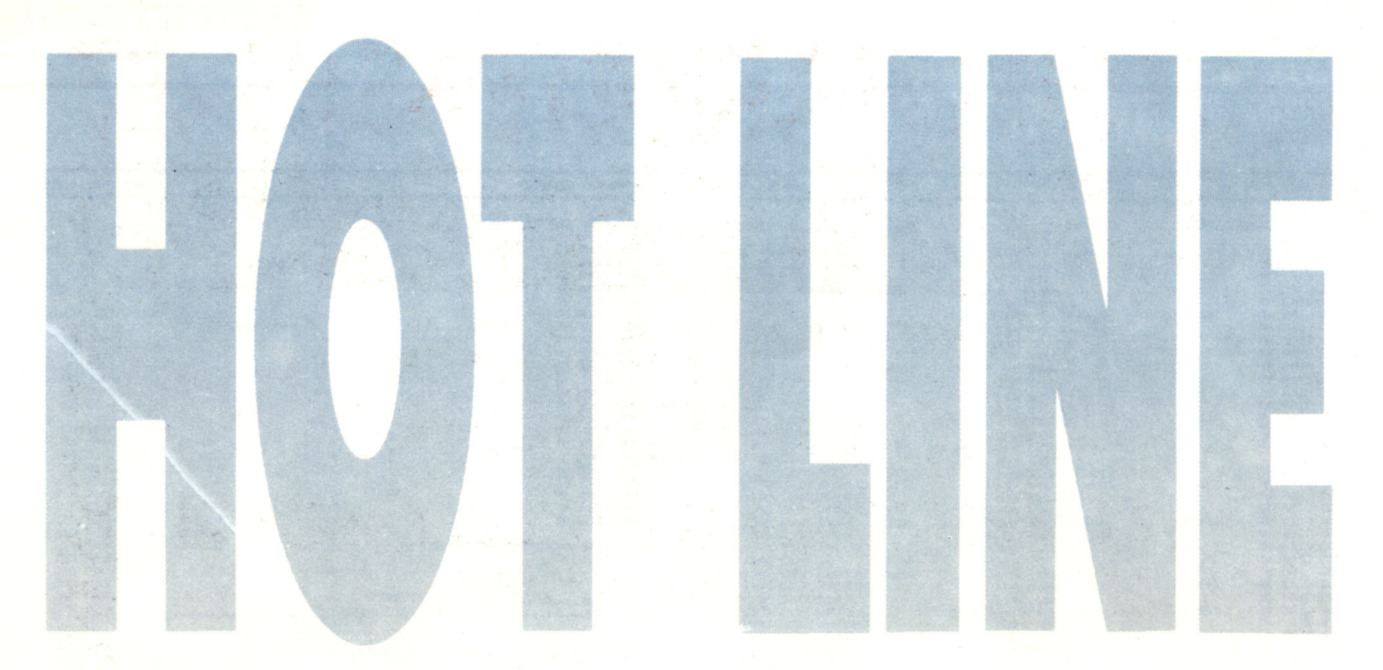
ASCII        $C14D  1610  1490  1530
BORRA        $E544   140   250
BUFFER       $033C   160   530   540   570   590   690   700   730   750
                           790   800   820   830  1040
CHROUT       $E10C   170  1500  1540
CIFRA        $C15A  1680  1620
CINCO        $C106  1290  1240
COLOR        $D020    90   230   240
CUATRO       $C0F8  1230  1180
DECODE       $C11C  1390   610   770   960
DOLAR        $C15E  1710   560   720   950
DOS          $C0DC  1110  1060
EDITCA       $AB1E   130   280   350   420   450   520   680   910  1030
                          1090  1150  1210  1270  1330  1370  1730
EDITNU       $BDCD   120   550   710   940
ERROR        $C114  1350  1300
FIN          $C11B  1380  1100  1160  1220  1280  1340
LIBRE3       $0334   100   580   740   810   930  1430  1450
LIBRE4       $0335   110   600   760   840   920  1390  1410
MAIN1        $C135  1480  1400  1440
MAIN2        $C13F  1520  1420  1460
SAL          $C15D  1700  1670
SETCUR       $E50A   150   320   390   490   650   880  1000
SHIFT        $C148  1560  1480
TEXT0        $C166  1750  1710  1720
TEXT1        $C169  1760   260   270
TEXT10       $C1EC  1850  1190  1200
TEXT11       $C1FE  1860  1250  1260
TEXT12       $C210  1870  1310  1320
TEXT13       $C21B  1880  1350  1360
TEXT2        $C18B  1770   330   340
TEXT3        $C1A4  1780   400   410
TEXT4        $C1AC  1790   500   510
TEXT5        $C1B4  1800   660   670
TEXT6        $C1BB  1810   890   900
TEXT7        $C1C7  1820  1010  1020
TEXT8        $C1CD  1830  1070  1080
TEXT9        $C1DC  1840  1130  1140
TRES         $C0EA  1170  1120


A LISTING, PAGE #9
========================================================================

CONCORDANCE TABLE IN NUMERICAL ORDER:

LABEL        VALUE DEFN. REFERENCES

LIBRE3       $0334   100   580   740   810   930  1430  1450
LIBRE4       $0335   110   600   760   840   920  1390  1410
BUFFER       $033C   160   530   540   570   590   690   700   730   750
                           790   800   820   830  1040
EDITCA       $AB1E   130   280   350   420   450   520   680   910  1030
                          1090  1150  1210  1270  1330  1370  1730
EDITNU       $BDCD   120   550   710   940
DOS          $C0DC  1110  1060
TRES         $C0EA  1170  1120
CUATRO       $C0F8  1230  1180
CINCO        $C106  1290  1240
ERROR        $C114  1350  1300
FIN          $C11B  1380  1100  1160  1220  1280  1340
DECODE       $C11C  1390   610   770   960
MAIN1        $C135  1480  1400  1440
MAIN2        $C13F  1520  1420  1460
SHIFT        $C148  1560  1480
ASCII        $C14D  1610  1490  1530
CIFRA        $C15A  1680  1620
SAL          $C15D  1700  1670
DOLAR        $C15E  1710   560   720   950
TEXT0        $C166  1750  1710  1720
TEXT1        $C169  1760   260   270
TEXT2        $C18B  1770   330   340
TEXT3        $C1A4  1780   400   410
TEXT4        $C1AC  1790   500   510
TEXT5        $C1B4  1800   660   670
TEXT6        $C1BB  1810   890   900
TEXT7        $C1C7  1820  1010  1020
TEXT8        $C1CD  1830  1070  1080
TEXT9        $C1DC  1840  1130  1140
TEXT10       $C1EC  1850  1190  1200
TEXT11       $C1FE  1860  1250  1260
TEXT12       $C210  1870  1310  1320
TEXT13       $C21B  1880  1350  1360
COLOR        $D020    90   230   240
CHROUT       $E10C   170  1500  1540
SETCUR       $E50A   150   320   390   490   650   880  1000
BORRA        $E544   140   250
```



Desde Alicante, nos escribe un reciente commodoriano, se trata de José Serer Catalá, que tiene las siguientes dudas:

1. ¿Es preciso desconectar el sistema Fuera Errores antes de grabar el programa tecleado?

2. ¿Ocurre algo si no se desconecta al ejecutar un programa?

3. ¿Hay que cargar el programa reloj (Octubre), cada vez que se vaya a usar con otro programa?

4. ¿Se queda grabado en los programas que se hayan podido grabar?

5. ¿Cómo se trabaja con los programas que tienen varios listados a la vez?

6. ¿Cómo se consigue la abreviatura de POKE?

1. Al terminar de teclear un programa de nuestra revista, para proceder a su grabación, es necesario desconectar el sistema Fuera Errores mediante la pulsación simultánea de las teclas RUN/STOP+RESTORE; sobre todo para los usuarios de Datassette. Esto se debe a que el programa Fuera Errores, se haya ubicado precisamente ahí, en el «Buffer del casete», que es por donde «circularán» los datos del programa al grabarse en la cinta. Para los usuarios de disco, esta operación no es tan crucial, pero se recomienda también su desconexión, debido a que Fuera Errores, usa interrupciones, que como norma general, se deberán desactivar SIEMPRE, antes de cualquier operación de Entrada/Salida.

2. Si el programa Fuera Errores sigue activado al ejecutar un programa, pueden pasar muchas cosas; desde que no pase nada, hasta que el ordenador se cuelgue. El que suceda una cosa u otra, vendrá determinado por las características del programa que se ejecute; si éste utiliza el mismo tipo de interrupción (IRQ) que el Fuera Errores, seguro que habrá problemas o si emplea la zona de memoria antes comentada (buffer del casete). En cualquier caso, y puesto que ya no se va a necesitar al ejecutar un programa, se DEBE desactivar siempre antes de probar otro programa.

3. Siempre que se quiera utilizar un programa de ampliación, como el reloj del número de octubre o incluso el mismo Fuera Errores, se debe cargar primero esta aplicación y ejecutarla (RUN), y a continuación, cargar el programa que funcionará «conjuntamente» con la aplicación.

4. Esta pregunta, está íntimamente relacionada con la anterior; la respuesta es NO, cuando se graba un programa que use de otro, como comodidad, información o por cualquier otro motivo, generalmente esta utilidad se haya escrita en código máquina, en posiciones de memoria que nada tienen que ver con el BASIC (por ejemplo el Fuera Errores en el buffer del casete), por lo que al grabar el programa, la aplicación (Fuera Errores, Reloj, etc.), no se grabará con él, teniendo que proceder a su carga, cada vez que se quiera emplear, como veíamos en la respuesta anterior. No obstante, si no hemos apagado el ordenador, y la aplicación no ha sido borrada por el programa utilizado, siempre cabe la posibilidad de activarla de nuevo (si está fuera de la zona BASIC como es lo habitual, incluso después de un NEW), mediante el SYS correspondiente.

5. La forma de trabajar con los programas que tienen varios listados a la vez, se explica detalladamente en el texto del artículo. En algunos casos, se trata de programas complementarios, en los cuales habrá que ejecutar el primero, después el segundo... etc.; en otros casos, será el primer programa (cargador), el que se ocupe de cargar al 0 a los siguientes, y otras veces se tratará de programas de ejemplo de un programa principal. Pero siempre, su uso quedará explicado en el texto del artículo que les acompaña.

6. Para abreviar la palabra POKE, deberás pulsar la letra P y a continuación mantener pulsada SHIFT con la letra O (pO). En general, las abreviaturas de las palabras clave, se consiguen pulsando la primera letra de la palabra más la segunda letra con SHIFT. Para más información, consulta las tablas situadas al final del manual del C64.

Juan Manuel Moyano Rodríguez (Málaga) tiene el problema en la frecuente aparición del mensaje de error «out of data» al teclear los programas que aparecen en TECLA a TECLA.

Como muy bien indica el texto del mensaje, este error se produce al intentar leer más datos de los disponibles dentro del programa, presentes dentro de las líneas **DATA** y son tratados por el intérprete como si fuesen una única lista SECUENCIAL. Este mensaje puede aparecer por dos causas fundamentales: la primera es que el rango del **FOR** que los lee es superior al número de datos presentes en estas líneas **DATA;** y la segunda, es que no haya suficiente número de datos debido a que te hayas «comido» alguna línea **DATA;** no obstante, también puede aparecer si te has «comido» alguna coma entre datos, lo cual produciría la confusión de dos o más datos como uno solo faltando lógicamente al final.

Para solucionar este problema, deberás usar el sistema Fuera Errores!, siguiendo atentamente las instrucciones de uso que aparecen en cada número de nuestra revista, ya que parece altamente improbable que el ordenador genere estos mensajes por un mal funcionamiento. Si siguen apareciendo los mensajes, vuelve a escribirnos aclarándonos en qué línea se producen y en qué programas.

PREMIOS MARKETING 86-87

RESULTADOS PROVISIONALES

INFORMATICA PERSONAL - TU MICRO AMSTRAD
TU MICRO COMMODORE

LA MEJOR CAMPAÑA DE PUBLICIDAD

Periféricos/Accesorios	Ordenadores	Programas
1. NEC (Omnilogic)	Amstrad (Indescomp)	Microbyte
2. IDEAssociates (Omnilogic)	Ericsson	Soft Express
3. DK'Tronics (Comercial Hernao)	Commodore Amiga (Microelectrónica y Control)	GEM (Casa de Software)
4. Ofites Informática	Bull	Dro Soft

EL MEJOR ANUNCIO

Periféricos/Accesorios	Ordenadores	Programas
1. NEC Pinwriter (Omnilogic)	Amstrad PC1512 (Indescomp)	MS Word (Intertec)
2. Soportes mag. (3M)	Amstrad PCW8512 (Indescomp)	Zombi (Soft Express)
3. Lápiz óptico (Ofites)	Gama Toshiba (Española de Microordenadores)	Turbo Prolog Borland (Micronet)
4. Kyocera (Dirac)		Word Perfect (Keylan)
		Mail Order Masters (Dro)

LA MEJOR GAMA DE PRODUCTO

Periféricos/Accesorios	Ordenadores	Programas
1. D.S.E.	Amstrad (Indescomp)	Micronet
2. Calcomp	Toshiba (Española de Microordenadores)	Mastertronic (Dro Soft)
3. DK'Tronics (Comercial Hernao)	Tandon	Logic Control
4. Ofites Informática	Commodore (Microelectrónica y Control)	Erbe Software

EL MEJOR PRODUCTO

Periféricos/Accesorios	Ordenadores	Programas
1. Pinwriter P5XL (Omnilogic)	IBM XT-286	Word Perfect
2. Conv. monitor/TV (DK'Tronics)	Amstrad PC1512 (Indescomp)	Lotus 1-2-3
3. Modem Nightingale (Agroinformática y Comunicaciones)	Compaq. Deskpro 386	Minja (Dro Soft)
4. Compacto Microgayma (Gayma)	Epson PC	Asphalt (Soft Express)

Las votaciones de los lectores han sido recogidas a través de las correspondientes tarjetas incluidas en los números de diciembre-86 y enero-87 de las revistas MICROINFORMATICA PERSONAL, TU MICRO AMSTRAD y TU MICRO COMMODORE de Ediciones INGELEK.



Alkerke

Los juegos de salón han sido siempre un gran entretenimiento para la humanidad. Sólo desde hace poco, los ordenadores domésticos van cambiando nuestra forma de jugar. Incluso así, aún sigue siendo entretenimiento pasar una tarde con el parchís o la oca... aunque no tanto como con el Alkerke.

En esta ocasión, os traemos un programa «de salón». ALKERKE está basado en el Alquerque de nueve, un juego oriental muy poco conocido, pero de gran dificultad y elegancia... si es que a hacer tres en raya se le puede llamar elegante.

El programa está pensado para dos participantes, y se juega sobre un tablero consistente en tres cuadrados unidos por la parte central de los lados. Cada jugador dispone de 9 fichas, que podrá colocar, por turno, en cada una de las intersecciones del tablero.

El objetivo es... ¡hacer tres en raya! (¿has dicho sencillo? Ya verás, ya...). Las tres en raya pueden conseguirse en vertical, horizontal, o en cada una de las líneas de unión de los cuadrados (las diagonales no valen). Cada vez que uno de los jugadores logra hacer tres en raya, puede quitarle al contrincante cualquiera de las fichas que tenga sobre el tablero; pierde el jugador que llegue a quedarse con dos fichas.

Pero... Hay un detalle: cada jugador no puede hacer tres en raya en la misma posición dos veces. Es decir, si por ejemplo conseguimos la jugada en el lado superior, ya no podemos volver a hacerlo en ese mismo lado (aunque nuestro adversario sí, si no lo ha hecho previamente). Las demás posiciones del tablero siguen siendo utilizables (y hay 16 posibles posiciones... ¡sobran!).



Un juego, dos partes...

El juego en sí puede considerarse dividido en dos fases: una primera, que llamaremos salida, en la cual los jugadores van poniendo las fichas sobre el tablero, y el juego propiamente dicho. En la primera fase, las fichas se depositan sobre una posición del tablero, no existiendo ninguna regla que limite su movilidad, y con la posibilidad, por supuesto, de hacer tres en raya... si el contrario lo permite. Lo único que está prohibido es mover una pieza del tablero hasta que se hayan sacado todas (y hay 9).

Cada jugador, alternativamente, pone una ficha. Esta fase es de gran importancia, pues a menudo, de ella depende después que un jugador esté en posición favorable para la defensa o el ataque (un consejo: evita las esquinas de los cuadros y busca las intersecciones con el cuadro central... De nada).

Al situar la última ficha, comienza la segunda fase, donde la materia gris a veces degenera a negro. Ahora sólo es posible mover las fichas dentro del tablero, sobre las líneas del mismo, pero atención: no podemos pasar por ninguna intersección que esté ocupada por otra ficha, nuestra o del adversario. La necesidad de buscar, ¡o de abrir!, un camino de una posición a otra, confiere al juego una alta dosis de estrategia o, si se recurre al martillo contra el contrario, de ejercicio físico.

Una vez que se toca una ficha, debe moverse, como en las partidas (serias) de ajedrez; en caso de no querer mover, podemos tocar una ficha y dejarla en su posición.

Y esas son todas las reglas; nada difíciles, como podemos ver.

Alkerke y un commodore

El paso del alquerque de nueve a ALKERKE es directo. Tendremos, en primer lugar, la opción de emplear joystick (uno en cada port para cada jugador, o bien uno sólo a compartir) o teclado (desgraciadamente, uno sólo a compartir), así como el color de las fichas de cada jugador.

A continuación, estaremos ante el tablero de juego. En la primera fase, una mano, del color del jugador que en ese momento deba mover, aparecerá en la zona central del mismo. Podemos desplazarla para colocar la ficha en la posición deseada, lo cual se consigue pulsando «fuego» en el joystick (o RETURN en el teclado).

Si se produce tres en raya, por supuesto, en una posición libre, el ordenador avisará, y permitirá al (por el momento) ganador retirar una de las fichas del contrario.

Posteriormente, ya en la fase de juego, la mecánica es la misma: movemos la mano hasta tocar la ficha a mover, y pulsamos «fuego» (o RETURN). Hecho esto, podemos mover nuestra ficha, pero, observaremos que ahora no es posible sacarla del tablero y que tozudamente se negará a pasar por encima de una compañera.

Los posibles errores (o intentos tímidos de falsificar el transcurso del juego, es decir, trampas) serán señalados por un aviso sonoro para ser ignorados a continuación.

Gana el jugador que primero consigue dejar a su rival con dos fichas. A continuación podremos elegir entre terminar o seguir con otra partida.

Y nada más; te dejamos para que tu habilidad y tu Commodore lleven la derrota a cualquier adversario alkerkiano al que te enfrentes...

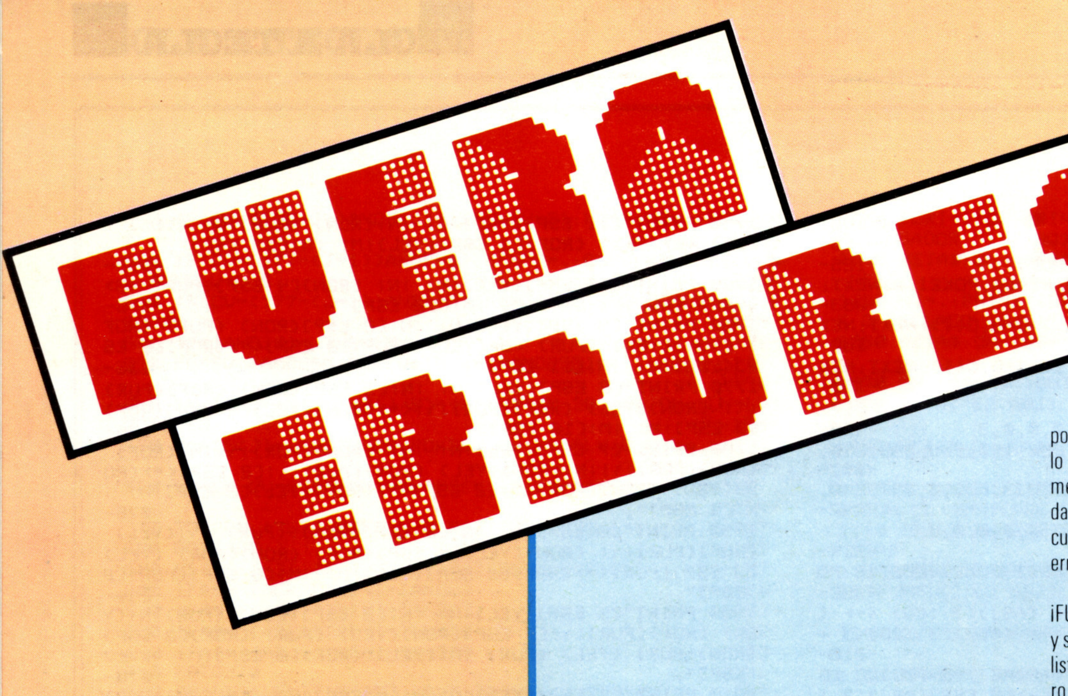
LISTADO

```
1 REM  ************************        -143-
2 REM *         ALKERKE         *       -233-
3 REM *       VERSION 1.1       *       -065-
4 REM *     (C) SINDIN SOFT     *       -193-
5 REM * 1987 TU MICRO COMMODORE *       -238-
6 REM  ************************        -148-
7 :                                     -145-
10 GOTO990                              -092-
20 IFLV=1147THENT(0,1)=MD:RETURN        -006-
30 IFLV=1156THENT(0,2)=MD:RETURN        -008-
40 IFLV=1165THENT(0,3)=MD:RETURN        -010-
50 IFLV=1507THENT(0,4)=MD:RETURN        -012-
60 IFLV=1525THENT(0,5)=MD:RETURN        -014-
70 IFLV=1867THENT(0,6)=MD:RETURN        -025-
80 IFLV=1876THENT(0,7)=MD:RETURN        -027-
90 IFLV=1885THENT(0,8)=MD:RETURN        -029-
100 IFLV=1270THENT(1,1)=MD:RETURN       -051-
110 IFLV=1276THENT(1,2)=MD:RETURN       -059-
120 IFLV=1282THENT(1,3)=MD:RETURN       -058-
130 IFLV=1510THENT(1,4)=MD:RETURN       -054-
140 IFLV=1522THENT(1,5)=MD:RETURN       -059-
150 IFLV=1750THENT(1,6)=MD:RETURN       -064-
160 IFLV=1756THENT(1,7)=MD:RETURN       -072-
170 IFLV=1762THENT(1,8)=MD:RETURN       -071-
180 IFLV=1393THENT(2,1)=MD:RETURN       -066-
190 IFLV=1396THENT(2,2)=MD:RETURN       -071-
200 IFLV=1399THENT(2,3)=MD:RETURN       -067-
210 IFLV=1513THENT(2,4)=MD:RETURN       -057-
220 IFLV=1519THENT(2,5)=MD:RETURN       -065-
230 IFLV=1633THENT(2,6)=MD:RETURN       -064-
240 IFLV=1636THENT(2,7)=MD:RETURN       -069-
250 IFLV=1639THENT(2,8)=MD:RETURN       -074-
260 FORII=1TO9                          -123-
270 PRINT"{HOM}{YEL}[6 ESP]FASE 1: SALIDA"II
                                        -005-
280 F=1:J=1:GOSUB650:GOSUB350:GOSUB660  -093-
290 F=1:J=2:GOSUB650:GOSUB350:GOSUB660  -095-
300 NEXT                                -242-
310 PRINT"{HOM}{YEL}[6 ESP]FASE 2: JUEGO[4 ESP]"
                                        -187-
320 F=2:J=1:GOSUB650:GOSUB350:GOSUB660  -089-
330 F=2:J=2:GOSUB650:GOSUB350:GOSUB660  -091-
340 GOTO310                             -132-
350 S(J)=-(F=1)                         -021-
360 ON-(NJ=.)GOTO590:I=PEEK(J(J))AND31  -000-
370 X=X-8*((IAND8)=.ANDX<300)+8*((IAND4)=.ANDX>25)
                                        -075-
380 Y=Y-8*((IAND2)=.ANDY<220)+8*((IAND1)=.ANDY>50)
                                        -070-
390 ON-(F=2ANDS(J)=1)GOTO400:GOTO450    -148-
400 L=PEEK(1024+INT((X-21)/8)+40*INT((Y-36)/8))
                                        -029-
410 IFL<>81THENIFL<>64THENIFL<>93THEN430 -023-
420 GOTO450                             -136-
430 X=X+8*((IAND8)=.ANDX<300)-8*((IAND4)=.ANDX>25)
                                        -072-
440 Y=Y+8*((IAND2)=.ANDY<220)-8*((IAND1)=.ANDY>50)
                                        -067-
450 ON-((IAND16)<>.ORSS=1)GOTO520:S(J)=-(S(J)=.ANDF<3
)                                       -153-
460 IFCV=81ANDS(J)=.ANDF<3THENMD=J:GOSUB20:GOSUB1590:
RETURN                                  -066-
470 IFCV<>81ANDS(J)=.ANDF<3THENS(J)=1:GOSUB1670:GOTO5
20                                      -151-
480 IF(CV<>(176+J))ANDS(J)=1ANDF<3THENS(J)=.:GOSUB167
0                                       -218-
```

```
490 IF(CV=(176+J))ANDS(J)=1THENCV=81:KV=5:MD=.:GOSUB2
0                                                    -019-
500 IF(CV=(177-(J=1))ANDF=3)THENMD=.:GOSUB20:POKELV,8
1:POKELV+O,5                                         -231-
510 IF(CV=(177-(J=1))ANDF=3)THENGOSUB1590:GOSUB1670:R
ETURN                                                -059-
520 IF(IAND16)=.THENSS=.                             -197-
530 POKEV,XAND255:POKEV+16,-((XAND256)<>.):POKEV+1,Y
                                                     -140-
540 LN=1024+INT((X-21)/8)+40*INT((Y-36)/8):ON-(LN=LV)
GOTO360                                              -154-
550 ON-(S(J)=.)GOTO560:POKELV,CV:POKELV+O,KV
                                                     -036-
560 LV=LN:CV=PEEK(LN):KV=PEEK(LN+O):IFS(J)=.THEN360
                                                     -224-
570 POKELN,176+J:POKELN+O,C(J):GOTO360               -087-
580 GOTO360                                          -143-
590 GETA$:I=31:IFA$=CHR$(13)THENI=IAND15             -127-
600 IFA$="{DCH}"THENI=IAND23                         -126-
610 IFA$="{IZQ}"THENI=IAND27                         -003-
620 IFA$="{ARB}"THENI=IAND30                         -242-
630 IFA$="{ABJ}"THENI=IAND29                         -123-
640 GOTO370                                          -141-
650 PRINT"{HOM}{15 ABJ}"SPC(30)N$(J):RETURN          -095-
660 Q=.:FORA=.TO2                                    -026-
670 IFT(A,1)=JTHENIFT(A,2)=JTHENIFT(A,3)=JTHENIFH(J,3
*A)=.THEN690                                         -205-
680 GOTO700                                          -142-
690 Q=3*A:A=3:GOTO790                                -233-
700 IFT(A,1)=JTHENIFT(A,4)=JTHENIFT(A,6)=JTHENIFH(J,3
*A+2)=.THEN720                                       -035-
710 GOTO730                                          -139-
720 Q=3*A+2:A=3:GOTO790                              -064-
730 IFT(A,6)=JTHENIFT(A,7)=JTHENIFT(A,8)=JTHENIFH(J,3
*A+1)=.THEN750                                       -050-
740 GOTO760                                          -145-
750 Q=3*A+1:A=3:GOTO790                              -066-
760 IFT(A,3)=JTHENIFT(A,5)=JTHENIFT(A,8)=JTHENIFH(J,3
*A+3)=.THEN780                                       -053-
770 GOTO790                                          -151-
780 Q=3*A+3:A=3:GOTO790                              -071-
790 NEXT                                             -255-
800 IFA>3THENH(J,Q)=1:F=3:GOSUB940:J=1-(J=1):GOSUB159
0:J=1-(J=1)                                          -022-
810 IFA>3THENGOSUB350:GOSUB960:ON-(F(1-(J=1))<3)GOTO1
620:RETURN                                           -120-
820 IFT(0,2)=JTHENIFT(1,2)=JTHENIFT(2,2)=JTHENIFH(J,1
2)=.THEN840                                          -092-
830 GOTO850                                          -145-
840 A=4:Q=12:GOTO800                                 -164-
850 IFT(0,7)=JTHENIFT(1,7)=JTHENIFT(2,7)=JTHENIFH(J,1
3)=.THEN870                                          -114-
860 GOTO880                                          -151-
870 A=4:Q=13:GOTO800                                 -168-
880 IFT(0,4)=JTHENIFT(1,4)=JTHENIFT(2,4)=JTHENIFH(J,1
4)=.THEN900                                          -103-
890 GOTO910                                          -148-
900 A=4:Q=14:GOTO800                                 -163-
910 IFT(0,5)=JTHENIFT(1,5)=JTHENIFT(2,5)=JTHENIFH(J,1
5)=.THEN930                                          -104-
920 RETURN                                           -155-
930 A=4:Q=15:GOTO800                                 -167-
940 PRINT"{HOM}{24 ABJ}[3 -] HAS HECHO 3 EN RAYA, JUG
ADOR"J"[3 -]{HOM}"                                   -022-
950 GOSUB1670:RETURN                                 -038-
960 PRINT"{HOM}{24 ABJ}[38 ESP]{HOM}"                -014-
970 A=1-(J=1):F(A)=F(A)-1:POKEP(A,1)+F(A),32:RETURN
                                                     -128-
980 RETURN                                           -161-
990 POKE53280,0:POKE53281,.:GOSUB1710:PRINT"{CLR}":V=
53248:O=54272:POKEV+29,1                             -133-
1000 DIMT(2,8),H(2,15),P(2,1):C(1)=5:C(2)=1:FORI=.TO2
:FORJ=1TO8:T(I,J)=.                                  -033-
1010 NEXT:NEXT:FORI=.TO2:FORJ=.TO15:H(I,J)=.:NEXT:NEX
T                                                    -018-
1020 P(1,.)=1099:P(2,.)=1259:P(1,1)=1171:P(2,1)=1331
                                                     -115-
1030 FORI=1TO2:PRINT"{8 ABJ}[6 ESP]{GRN}{RON} COLOR D
EL JUGADOR "I"{IZQ} (2/8)? "                         -077-
1040 GETA$:ON-(A$<"2"ORA$>"8")GOTO1040:C(I)=VAL(A$)-1
:NEXT                                                -024-
1050 NJ=.:PRINT"{CLR}{8 ABJ}[6 ESP]{GRN}{RON} JOYSTIC
K O TECLADO (J/T)? "                                 -183-
1060 GETA$:ON-(A$="J"ORA$="T")GOTO1070:GOTO1060
                                                     -199-
1070 ON-(A$="T")GOTO1170                             -063-
1080 PRINT"{2 ABJ}[11 ESP]{GRN}{RON} JOYSTICKS (1/2)
"                                                    -114-
1090 GETA$:ON-(A$="1"ORA$="2")GOTO1100:GOTO1090
                                                     -140-
1100 NJ=VAL(A$):PRINT"{2 ABJ}{RON}[5 ESP]MOVED LA MAN
O CON EL JOYSTICK,[5 ESP]";                          -234-
1110 PRINT"[4 ESP]<FUEGO> PARA COGER / DEJAR FICHA.[3
 ESP]"                                               -214-
1120 IFNJ>1THEN1190                                  -116-
1130 PRINT"{2 ABJ}[14 ESP]{GRN}{RON} PORT (1/2) "
                                                     -080-
1140 GETA$:ON-(A$="1"ORA$="2")GOTO1150:GOTO1140
                                                     -137-
1150 IFA$="1"THENJ(1)=56321:J(2)=J(1):GOTO1200
                                                     -012-
1160 J(1)=56320:J(2)=J(1):GOTO1200                   -055-
1170 PRINT"{2 ABJ}{RON}USA TECLAS DE CURSOR PARA MOVE
R LA MANO,";                                         -166-
1180 PRINT"[3 ESP]<RETURN> PARA COGER / DEJAR FICHA.[
3 ESP]"                                              -039-
1190 IFNJ=2THENJ(1)=56321:J(2)=56320                 -254-
1200 PRINT"{2 ABJ}[11 ESP]{GRN}{RON} PULSA UNA TECLA
":POKE198,.:WAIT198,1                                -139-
1210 PRINT"{CLR}{GRN}[24 ESP]!U!!13 *!!I! ";
                                                     -031-
1220 PRINT"[24 ESP]!-!JUGADOR 1:{RON}[3 ESP]{ROF}!-!
";                                                   -102-
1230 PRINT"[24 ESP]<Q>!13 *!<W> ";                   -048-
1240 PRINT"[3 ESP]!Q!!8 *!!Q!!8 *!!Q![2 ESP]!-![2 ESP
]{RON}123456789{ROF}[2 ESP]!-! ";                    -225-
1250 PRINT"[3 ESP]!-![8 ESP]!-![8 ESP]!-![2 ESP]<Q>!1
3 *!<W> ";                                           -105-
1260 PRINT"[3 ESP]!-![8 ESP]!-![8 ESP]!-![2 ESP]!-!JU
GADOR 2:{RON}[3 ESP]{ROF}!-! ";                      -162-
1270 PRINT"[3 ESP]!-![2 ESP]!Q!!5 *!!Q!!5 *!!Q![2 ESP
]!-![2 ESP]<Q>!13 *!<W> ";                           -001-
1280 PRINT"[3 ESP]!-![2 ESP]!-![2 ESP]!-![5 ESP]!-![5 ESP]!-![2
 ESP]!-![2 ESP]!-![2 ESP]{RON}123456789{ROF}[2 ESP]!-
! ";                                                 -131-
1290 PRINT"[3 ESP]!-![2 ESP]!-![5 ESP]!-![5 ESP]!-![2
 ESP]!-![2 ESP]!J!!13 *!!K! ";                       -030-
1300 PRINT"[3 ESP]!-![2 ESP]!-![2 ESP]!Q!!2 *!!Q!!2 *
!!Q![2 ESP]!-![2 ESP]!-![18 ESP]";                   -119-
1310 PRINT"[3 ESP]!-![2 ESP]!-![2 ESP]!-![5 ESP]!-![2
 ESP]!-![2 ESP]!-![18 ESP]";                         -095-
1320 PRINT"[3 ESP]!-![2 ESP]!-![2 ESP]!-![5 ESP]!-![2
 ESP]!-![2 ESP]!-![2 ESP]!U!!13 *!!I! ";             -222-
1330 PRINT"[3 ESP]!Q!!2 *!!Q!!2 *!!Q![5 ESP]!Q!!2 *!!
Q!!2 *!!Q![2 ESP]!-![3 ESP]JUGANDO[3 ESP]!-! ";
                                                     -187-
1340 PRINT"[3 ESP]!-![2 ESP]!-![2 ESP]!-![5 ESP]!-![2
 ESP]!-![2 ESP]!-![2 ESP]<Q>!13 *!<W> ";             -160-
1350 PRINT"[3 ESP]!-![2 ESP]!-![2 ESP]!-![5 ESP]!-![2
 ESP]!-![2 ESP]!-![2 ESP]!-![13 ESP]!-! ";           -221-
1360 PRINT"[3 ESP]!-![2 ESP]!-![2 ESP]!Q!!2 *!!Q!!2 *
!!Q![2 ESP]!-![2 ESP]!-![2 ESP]!-![5 ESP]<2 +>[6 ESP]
!-! ";                                               -003-
1370 PRINT"[3 ESP]!-![2 ESP]!-![5 ESP]!-![5 ESP]!-![2
 ESP]!-![2 ESP]!-![6 ESP]<+>[6 ESP]!-! ";            -168-
1380 PRINT"[3 ESP]!-![2 ESP]!-![5 ESP]!-![5 ESP]!-![2
 ESP]!-![2 ESP]!-![6 ESP]<+>[6 ESP]!-! ";            -169-
1390 PRINT"[3 ESP]!-![2 ESP]!Q!!5 *!!Q!!5 *!!Q![2 ESP
]!-![2 ESP]!-![6 ESP]<+>[6 ESP]!-! ";                -198-
1400 PRINT"[3 ESP]!-![8 ESP]!-![8 ESP]!-![2 ESP]!-![6
 ESP]<+>[6 ESP]!-! ";                                -040-
1410 PRINT"[3 ESP]!-![8 ESP]!-![8 ESP]!-![2 ESP]!-![5
 ESP]<3 +>[5 ESP]!-! ";                              -053-
1420 PRINT"[3 ESP]!Q!!8 *!!Q!!8 *!!Q![2 ESP]!-![13 ES
```

```
P]¦-¦ ";                                                -128-
1430 PRINT"[24 ESP]¦-¦[13 ESP]¦-¦ ";                    -110-
1440 PRINT"[24 ESP]¦J¦¦13 *¦¦K¦ {HOM}";                 -125-
1450 F(1)=9:F(2)=9:FORI=832TOI+63:READA:POKEI,A:NEXT:
POKE2040,13                                             -083-
1460 FORI=1TO2:FORJ=.TO2:POKEP(I,.)+J+O,C(I):NEXT:NEX
T                                                       -160-
1470 FORI=1TO2:FORJ=.TO8:POKEP(I,1)+J+O,C(I):NEXT:NEX
T                                                       -170-
1480 J=2:GOSUB1590:POKEV+21,1                           -217-
1490 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,3                -159-
1500 DATA240,0,31,255,128,95,255,252,190,255,255,240,
15,255,64,119                                           -007-
1510 DATA255,0,233,255,1,208,255,1,233,255,0,231,240,
0,95,192                                                -247-
1520 DATA0,63,128,0,62,0,0,0,0,0,0,0,0,0,0,0
                                                        -114-
1530 FORIJ=54272TO54295:POKEIJ,0:NEXT:POKE54296,15:PO
KE54277,8                                               -252-
1540 N$(1)="<2 +> {ABJ}{3 IZQ} <+> {ABJ}{3 IZQ} <+> {
ABJ}{3 IZQ} <+>{DCH}{ABJ}{3 IZQ} <+> {ABJ}{3 IZQ}<3 +
>"                                                      -218-
1550 N$(2)=" <+> {ABJ}{3 IZQ}<+> <+>{ABJ}{3 IZQ}[2 ES
P]<+>{ABJ}{3 IZQ} <+> {ABJ}{3 IZQ}<+>[2 ESP]{ABJ}{3 I
ZQ}<3 +>"                                               -223-
1560 LV=1024+(X-21)/8+40*INT((Y-40)/8):CV=PEEK(LV)
                                                        -028-
1570 KV=PEEK(LV+O)                                      -093-
1580 GOTO260                                            -191-
1590 CE=C(2):X=96:Y=176:IFJ=2THENY=104:CE=C(1)
                                                        -198-
1600 POKEV,.:POKEV+1,Y:POKEV+16,-((XAND256)<>.):POKEV
+39,CE                                                  -044-
1610 LV=1024+(X-21)/8+40*INT((Y-40)/8):CV=PEEK(LV):KV
=PEEK(LV+O):RETURN                                      -220-
1620 POKEV+21,.:PRINT"{CLR}{PUR} [6 -] HA VENCIDO EL
JUGADOR [6 -] "                                         -186-
1630 PRINT"{HOM}{5 ABJ}"SPC(17)N$(J)                    -207-
1640 PRINT"{2 ABJ}[7 ESP]{WHT}DESEAIS JUGAR MAS (S/N)
?"                                                      -219-
1650 GETA$:IFA$="S"THENRUN                              -087-
1660 ON-(A$<>"N")GOTO1650:PRINT"{CLR}PUES ADIOS!":END
                                                        -027-
1670 FORIJ=54272TO54295:POKEIJ,0:NEXT:POKE54296,15:PO
KE54277,8                                               -001-
1680 POKE54278,255:POKE54276,23:LO=2:FORIJ=1TO3:LH=8:
POKE54273,LO                                            -166-
1690 FORIK=1TO3:POKE54287,LH:LH=LH*1.6:NEXT:LO=LO*1.9
:NEXT                                                   -161-
1700 FORIK=1TO5:POKE54287,LH:LH=LH*1.08:NEXT:POKE5427
8,15:RETURN                                             -004-
1710 PRINT"{CLR}[8 ESP]{RON}{RED}SINDIN SOFT PRESENTA
[3 .]{ROF}[9 ESP]";                                     -149-
1720 PRINT"{ABJ}[13 ESP]{YEL}¦N¦<G>{RON}{PUR}¦\¦ ¦N¦{
ROF} {RON}{PUR}¦M¦<*>{ROF}[2 ESP]{YEL}<M>¦M¦[15 ESP]"
;                                                       -205-
1730 PRINT"[12 ESP]{YEL}¦N¦<M>{RON}{PUR}¦\¦{YEL}[2 ES
P]{PUR}¦2 N¦<G>{YEL} {ROF}[3 ESP]{YEL}<G>¦M¦[14 ESP]"
;                                                       -028-
1740 PRINT"[11 ESP]{YEL}¦N¦ ¦N¦{RON} {PUR}¦N¦ {YEL} {
PUR} <G>{YEL} {ROF}[3 ESP]{YEL}¦M¦ ¦M¦[13 ESP]";
                                                        -053-
1750 PRINT"[10 ESP]{YEL}¦N¦ ¦N¦ {RON}[4 ESP]{PUR} <G>
{YEL} {ROF}[4 ESP]¦M¦ ¦M¦[12 ESP]";                     -176-
1760 PRINT"[9 ESP]{YEL}¦N¦ ¦N¦[2 ESP]{RON} {PUR} {ROF
} {RON}{YEL} {PUR} <G>{YEL} {PUR}[2 ESP]<*>{ROF}[2 ES
P]{YEL}¦M¦ ¦M¦[11 ESP]";                                -136-
1770 PRINT"[8 ESP]{YEL}¦N¦ ¦N¦[3 ESP]{RON} {ROF}{PUR}
¦\¦ {RON}{YEL} {ROF}{PUR}¦\¦<*>{RON}{YEL}[4 ESP]{ROF}
[3 ESP]¦M¦ ¦M¦[10 ESP]";                                -128-
1780 PRINT"[7 ESP]{YEL}¦N¦ ¦N¦ {RON}{PUR}¦\¦¦N¦{ROF}
{RON}{PUR}¦\¦¦N¦{ROF}{YEL} {RON}{PUR}¦\¦[2 ESP]<*>{RO
F}{YEL} {RON}{PUR}¦M¦[2 ESP]<*>{ROF}{YEL}[2 ESP]¦M¦ ¦
M¦[9 ESP]";                                             -076-
1790 PRINT"[6 ESP]¦N¦ ¦N¦[2 ESP]{RON} {PUR} ¦\¦{YEL}
{ROF}{PUR}¦\¦ {RON}{YEL}[4 ESP]{ROF} {RON}{PUR} {YEL}
[3 ESP]{PUR}<*>{ROF}[2 ESP]{YEL}¦M¦ ¦M¦[6 ESP]{PUR}[2
 ESP]";                                                 -244-
1800 PRINT"[5 ESP]{YEL}<M> ¦N¦[3 ESP]{RON} {PUR}¦N¦{Y
EL} {ROF}{PUR}¦\¦[2 ESP]{RON}{YEL} {PUR} {ROF}[3 ESP]
{RON}{PUR} {YEL} {PUR} ¦M¦{YEL} {ROF}[3 ESP]¦M¦ <G>[7
 ESP]";                                                 -078-
1810 PRINT"[5 ESP]{YEL}<M>¦M¦[4 ESP]{RON}[2 ESP]{PUR}
¦2 N¦{ROF}[2 ESP]{RON}{YEL}[2 ESP]{ROF}[3 ESP]{RON}{P
UR} {YEL}[3 ESP]{ROF}[5 ESP]¦N¦<G>[7 ESP]";             -183-
1820 PRINT"[5 ESP]<M> ¦M¦[3 ESP]{RON}{YEL} {PUR} {YEL
} {PUR}¦2 N¦{ROF} {RON}{YEL} {PUR}¦N¦ <*>{ROF} {RON}{
PUR} {YEL} {ROF}{PUR}<*>{RON}{YEL} {PUR}<*>{ROF}[3 ES
P]{YEL}¦N¦ <G>[7 ESP]";                                 -145-
1830 PRINT"[6 ESP]¦M¦ ¦M¦[2 ESP]{RON} {ROF}{PUR}¦\¦ {
RON}{YEL} {ROF}{PUR}¦\¦ {RON}{YEL}[4 ESP]{ROF} {PUR}<
*>{RON}{YEL} {ROF} {PUR}<*>{RON}{YEL} {ROF}[2 ESP]{YE
L}¦N¦ ¦N¦[8 ESP]";                                      -248-
1840 PRINT"[7 ESP]{YEL}¦M¦ ¦M¦[4 ESP]{RON}{PUR}¦\¦¦N¦
{ROF} {RON}{PUR}¦\¦¦N¦¦M¦[3 ESP]<*>{ROF}[4 ESP]{YEL}¦
N¦ ¦N¦[9 ESP]";                                         -069-
1850 PRINT"[8 ESP]¦M¦ ¦M¦[3 ESP]{RON}{YEL} {PUR} ¦\¦{
YEL} {ROF}{PUR}¦\¦{RON}<G>{YEL}[4 ESP]{ROF}[3 ESP]¦N¦
 ¦N¦[10 ESP]";                                         -161-
1860 PRINT"[9 ESP]¦M¦ ¦M¦[2 ESP]{RON} {PUR}¦N¦{YEL} {
ROF}{PUR}¦\¦ {RON}{PUR}<G>{YEL} {PUR}<*>{ROF}[4 ESP]{
YEL}¦N¦ ¦N¦[11 ESP]";                                   -190-
1870 PRINT"[10 ESP]¦M¦ ¦M¦ {RON}[2 ESP]{PUR}¦2 N¦{ROF
} {RON}{PUR}<G>{YEL}[2 ESP]{ROF}[3 ESP]¦N¦ ¦N¦[12 ESP
]";                                                     -177-
1880 PRINT"[11 ESP]¦M¦ <G>{RON} {PUR} {YEL} {PUR}¦2 N
¦<G>{YEL} {PUR}[2 ESP]<*>{ROF}{YEL}<M> ¦N¦[13 ESP]";
                                                        -086-
1890 PRINT"[12 ESP]¦M¦<G>{RON} {ROF}{PUR}¦\¦ {RON}{YE
L} {ROF}{PUR}¦\¦<*>{RON}{YEL}[4 ESP]{ROF}<M>¦N¦[14 ES
P]{ABJ}";                                               -167-
1900 PRINT"[6 ESP](C) 1987 - TU MICRO COMMODORE[5 ESP
]{ABJ}";                                                -105-
1910 PRINT"[6 ESP]{RON}{GRN}PULSA UNA TECLA PARA EMPE
ZAR.{ROF}[5 ESP]{HOM}";                                 -228-
1920 POKE198,.:WAIT 198,1:POKE198,.:RETURN              -036-

READY.                                                  -173-
```



TU MICRO COMMODORE ha creado el revolucionario sistema de introducción de programas FUERA ERRORES. Este nos permitirá introducir, sin temor alguno al esfuerzo inútil, cualquier listado por largo y complicado que parezca.

Para adoptar los listados publicados bajo este sistema, deberemos seguir las siguientes normas:

1) Es fundamental transcribir EXACTAMENTE el listado reproducido, incluyendo todos sus espacios, aunque se trate de separaciones entre número de instrucción y línea de instrucción.

2) Todas las líneas finalizarán con un número de tres dígitos, encerrado entre guiones, que NO deberá ser introducido, puesto que no forma parte del programa, sino que tiene la finalidad de hacer funcionar el sistema FUERA ERRORES, según veremos más adelante. Para evitar equivocaciones, dicha cifra entre guiones se sitúa en el margen derecho del final de la línea BASIC a la cual corresponde, a una distancia prudencial del mismo.

3) Para facilitar la introducción de símbolos difícilmente interpretables, se procede a la siguiente representación en los listados.

– Las letras aparecidas entre menor y mayor deberán ser introducidas con pulsación simultánea de la tecla COMMODORE y la letra representada. Ej.: < M > =COMMODORE M.

– Las letras aparecidas entre barras verticales deberán ser introducidas como pulsación simultánea de la tecla SHIFT y la letra representada. Ej.:|K|= SHIFT K.

– Entre corchetes simples se representarán los símbolos que se obtienen por pulsación directa de la tecla, aunque lógicamente, este caso sólo se dará para indicar las sucesiones de más de una letra. Así por ejemplo, la introducción de 5 asteriscos se representaría por [5*].

– Para la repetición de símbolos obtenidos mediante las teclas COMMODORE o SHIFT, se seguirá una combinación de las tres normas anteriormente citadas. Así por ejemplo, la introducción de 10 símbolos COMMODORE H, se representaría por [< 10 H >].

– Para evitar confusiones, cuando se utilice el sistema de representación de sucesiones de carácter, y éste sea un espacio, se utilizará la abreviatura ESP. [15 ESP] = 15 espacios.

– Los caracteres de control, tales como desplazamientos del cursor, colores, estados de reversa y funciones, se simbolizarán por una abreviatura de tres letras (dos más un espacio en el caso de las funciones) encerradas entre llaves, tal como se señala en la tabla adjunta.

Para introducir cualquier listado por el sistema FUERA ERRORES, deberemos entrar previamente y ejecutar el listado BASIC que aparece en la página siguiente por lo cual es recomendable conservar una copia grabada del mismo, para sucesivas ocasiones. Una vez introducido este listado, ya sea por el teclado, o a través de cinta o disco, debemos ejecutarlo con **RUN**. Instantes más tarde aparecerá en la pantalla el mensaje **FUERA ERRORES!** y el cursor libre para la entrada de programas, con el tradicional **READY**. Por encima, lo cual indicará la activación del sistema de depuración de errores.

En virtud al **NEW** que finaliza línea 20 del programa ¡FUERA ERRORES!, éste habrá desaparecido de la memoria, y seremos libres para introducir cualquiera de los programas listados en la sección **TECLA A TECLA** de cualquier número de nuestra revista, o incluso aquellos que apareciendo en otras secciones se acojan a este sistema. Así pues, si el programa ¡FUERA ERRORES! ha desaparecido de la memoria, ¿qué hemos conseguido ejecutándolo? Bien, la respuesta se llama informáticamente **INTERRUPCION**: se trata de una técnica de programación en código máquina que permite que el ordenador efectúe prácticamente dos trabajos a un tiempo, o más correctamente, que ejecute determinada tarea de forma automática, sin necesidad de que le prestemos una atención constante, de forma similar al proceso de respiración en un humano.

Efectivamente, aunque el soporte BASIC ha desaparecido de la memoria, antes de ‹marcharse› ha dejado funcionando en modo interrupción la pequeña rutina en código máquina que se hallaba en sus **DATAS**. Para comprobarlo pulsa **RETURN**; observarás algo muy extraño: tu ordenador no se comporta normalmente, no sólo desciende una línea el cursor, sino que además hace aparecer un número en la esquina superior izquierda de la pantalla. A continuación veremos como emplearlo.

Cada vez que pulsemos **RETURN**, aparecerá un número en la mencionada zona de la pantalla, y éste corresponderá con la instrucción que hayamos introducido. Esto forma parte del sistema de FUERA ERRORES. Cuando introduzcamos cualquier línea de un listado de este tipo, deberemos fijarnos en el número que aparece al pulsar **RETURN** de fin de línea; si éste coincide con el que aparece en el listado al final de la línea, ésta habrá sido introducida correctamente, en caso contrario existe algún error de tecleado que debemos modificar. Para modificar una instrucción errónea, no tenemos ni tan siquiera que volver a teclearla si no queremos, bastará sencillamente con modificar el carácter o caracteres erróneos como siempre hacemos, hasta que coincida el número de verificación que se presentará al pulsar **RETURN**.

Así pues, el sistema FUERA ERRORES se compone de dos partes: una codificación especial de los listados que facilitan su introducción, evitando los errores al confundir los caracteres gráficos, de control, etc., y un sistema de verificación de líneas que nos advierte en el preciso instante de introducir una de estas, que está mal tecleada.

Ahora bien, el empleo de estos dos sistemas no quiere decir que nos encontremos ante un BASIC diferente al de COMMODORE 64. Este no ha cambiado, hace exactamente las mismas cosas de siempre; simplemente hemos cambiado la forma de hacer los listados. En cuanto al misterioso número que aparece en la esquina de la pantalla no es más que una simple suma de comprobación, lo que se conoce técnicamente como un CHECKSUM. La rutina en código máquina de interrupción suma los valores de los caracteres que entra-

TABLA DE INTERPRETACION DE CODIGOS DE CONTROL

ABR	SIGNIFICADO	OBTENCION
HOM	HOME	CLR/HOME
CLR	CLEAR + HOME	SHIFT CLR/HOME
ABJ	CURSOR ABAJO	CRSR VERTICAL
ARB	CURSOR ARRIBA	SHIFT CRSR VERTICAL
DCH	CURSOR DERECHA	CRSR HORIZONTAL
IZQ	CURSOR IZQUIERDA	SHIFT CSRS HORIZONTAL
RON	REVERSE ON	CTRL 9
ROF	REVERSE OFF	CTRL 0
F1	FUNCION 1	F1
F2	FUNCION 2	SHIFT F1
F3	FUNCION 3	F3
F4	FUNCION 4	SHIFT F3
F5	FUNCION 5	F5
F6	FUNCION 6	SHIFT F5
F7	FUNCION 7	F7
F8	FUNCION 8	SHIFT F7
BLK	BLACK (NEGRO)	CTRL 1
WHT	WHITE (BLANCO)	CTRL 2
RED	RED (ROJO)	CTRL 3
CYN	CYAN (CIAN)	CTRL 4
PUR	PURPPLE (PURPURA)	CTRL 5
GRN	GREEN (VERDE)	CTRL 6
BLU	BLUE (AZUL)	CTRL 7
YEL	YELLOW (AMARILLO)	CTRL 8
NRJ	NARANJA	COMMODORE 1
MRN	MARRON	COMMODORE 2
RCL	ROJO CLARO	COMMODORE 3
GR1	GRIS 1	COMMODORE 4
GR2	GRIS 2	COMMODORE 5
VCL	VERDE CLARO	COMMODORE 6
ACL	AZUL CLARO	COMMODORE 7
GR3	GRIS 3	COMMODORE 8
PI	π	SHIFT ↑
^	↑	↑
\	£	£
STP	STOP	RUN/STOP
INS	INSERT	INST/DEL
—	←	←

mos en cada línea que introducimos, aplicándoles un módulo 256, es decir, volviendo a cero cada vez que su suma parcial supera el 255, de este modo se crea un número entre 0 y 255 dependiente directamente de los caracteres introducidos. Así pues, es prácticamente imposible que una sucesión de errores den por casualidad ese número, mientras que siempre que la línea esté correctamente introducida obtendremos el mismo código que nosotros hallamos por igual sistema y añadimos al final de cada línea al realizar el listado del programa.

Por tanto, el misterioso número no es tampoco una modificación del BASIC de COMMODORE, sino simplemente un pequeño truco para la comprobación de que las líneas han sido bien introducidas. Propiamente no nos evita cometer errores de tecleado, sino que simplemente nos advierte inmediatamente en qué línea los hemos introducido.

Para desactivar el sistema sólo deberemos pulsar **RUN/STOP+RESTORE**, y si por cualquier motivo nos interesara reactivarlo, podríamos ejecutar **SYS 822**, siempre y cuando se encuentre el código máquina en la memoria, lógicamente.

¡ADVERTENCIA! Puesto que el código máquina se encuentra ubicado en el buffer del casete, es imprescindible desactivarlo **(RUN/STOP+RESTORE)** antes de realizar cualquier operación con dicho periférico.

Utilización del casete con fuera errores

Si queremos introducir parte de un programa, para continuar posteriormente el trabajo emprendido, sin perder por supuesto la enorme ventaja del FUERA ERRORES deberemos proceder de la siguiente forma:

1. Desactivar el sistema FUERA ERRORES mediante la pulsación de RUN/STOP + RESTORE.
2. Trasladar el código máquina desde su ubicación en el buffer del casete hasta otro punto desocupado en la memoria RAM (por ejemplo 49152). Para ello sólo hemos de saber que ocupa 114 bytes desde 822 (inclusive). Una sencilla rutina que efectúe el trabajo de reubicación puede ser: **FORI=0TO113:POKE49152+I,PEEK(822+I):NEXT.**
3. Realizar la operación correspondiente con el casete; ya sea grabar una copia de seguridad de lo introducido en la memoria hasta el momento, ya sea cargar el casete una parte ya grabada del programa.
4. Restablecer FUERA ERRORES a su punto original. Siguiendo el caso de la línea de ejemplo anterior: **FORI=0TO113:POKE822+I,PEEK(49152+I):NEXT.**
5. Reactivar el sistema FUERA ERRORES mediante SYS 822 (esta vez no aparecerá mensaje alguno, aunque al pulsar **RETURN** comprobaremos que aparecen las cifras de control en la esquina superior izquierda). Lógicamente, los pasos 4 y 5 no son necesarios si después de una grabación se va a apagar el ordenador, y sólo se llevarán a cabo después de las cargas, o si tras la grabación de una copia de seguridad del programa, deseamos continuar introduciéndolo acto seguido.

```
10 FORI=822TO935:READA:C=C+A:POKEI,A:NEXT
20 IFC<>15254THENPRINTCHR$(147)"ATENCION!, HAY UN ERROR EN LOS DATOS":END
30 PRINTCHR$(147)TAB(213)"FUERA ERRORES!":SYS822:NEW
100 DATA 169,3,141,37,3,169,69,141,36,3,169,0,133,254,96,32,87,241,133,251
110 DATA 134,252,132,253,8,201,13,240,13,24,101,254,133,254,165,251,166,252
120 DATA 164,253,40,96,169,13,32,210,255,165,214,141,176,3,206,176,3,169,0
130 DATA 133,216,169,18,32,210,255,169,19,32,210,255,169,45,32,210,255,166
140 DATA 254,224,100,176,5,169,48,32,210,255,224,10,176,5,169,48,32,210,255
150 DATA 169,0,133,254,32,205,189,169,45,32,210,255,173,176,3,133,214,76,88,3
```

TABLON

COMMODOE 64K, disquete 1541, IMP, RITMAN cassette, CN manuales, programas, cassette y disco (90.000 ptas). Dirigirse a Joan Palomas Mandoli. Canigo 12 Vic (Barcelona).

ME gustaría intercambiar programas en cinta para el C-64. Contestaré a todas las cartas. Interesados escribir a: Alberto Marote Liébana. De la Fé 4, 3.º 1.ª. 17300 Blanes (Gerona).

INTERCAMBIAMOS programas para C-128 y C-64 en disco. Club Commodore Canario. Apartado 2.485. 35080 Las Palmas de Gran Canaria.

VENDO 10 interfaces copiadores de seguridad a 3.000 ptas. cada uno. Ofrezo 600 programas por Unidad de Disco 1541. Llamar a partir de las 21 horas. Tel.: (91) 895 32 54. José Julio.

VENDO programa original Basic Lightning con instrucciones por tan sólo 3.000 ptas. regalo 10 programas de juego, también intercambio programas. Dirigirse a: José Ferrán González. Apartado 209. 06800 Mérida (Badajoz)

INTERCAMBIO programas en disco para C-64, poseo más de 1.000, prometo contestar. Mandar lista a Antonio Molina García. Barriada de la Constitución, bloque 12, 2.º C. Melilla

ATENCION urge comprar unidad de discos. O puedo ofrecer un lote de programas valorado en 130.000 ptas. a cambio de unidad de discos o de impresora o de monitor color de cualquier periférico para CBM-64. Cambio CBM-64 por CBM-128 pagando la diferencia y regalo programas. Vendo videopac Philips G-7400 convertible en ordenador, regalo programas. Intercambio software para CBM, tengo todas o casi todas las últimas novedades. Francisco Javier Bernal Malillos. Echegaray, 1. Cantalejo (Segovia). Tel.: (911) 52 04 04.

SE ha formado en Barcelona el club B.C.S, (Barna Computer Service) abarca Spectrum, Commodore, Amstrad y MSX, interesados escribir al Apartado 2309,08080 Barcelona, indicando modelo de ordenador o llamar Tel.: (93) 309 56 52 preguntar por Angel.

VENDO C-64 y unidad disco 1541 con poco uso en 65.000 ptas., regalo joystick Quickshot II y cinco discos con juegos. Bruno Celaya Montero. Larraina, 19-5 D. (Pamplona). Tel.: (948) 26 49 81.

VENDO juegos originales C-64 a 500 y 1.000 ptas. calidad de Kane o Green Beret; interesados facilito lista de los mismos. Bruno Celaya Montero. Larraina, 19-5 D (Pamplona). Tel.: (948) 26 49 81.

COMPRO originales para Commodore y Spectrum, compro revistas, sobre todo extranjeras y cambio programas para Commodore y Spectrum, Jorge Ron López. Angel Guimerá 36. Tel.: (93) 204 14 51. 08017 Barcelona.

DESEARIA contactar con usuarios del Commodore-64 para intercambiar programas, ya sean juegos, utilidades, trucos, etc. Interesados escribir a Víctor Felices. Pau Duarri 75, 3º 4ª. 08670 Navas (Barcelona). Mandar lista.

VENDO Commodore 128, unidad de disco 1571, monitor Philips 12" f. verde (cable incluido). Regalo libros, revistas, más de 10 discos con programas, etc. Dos meses de uso. Todavía en garantía. También por separado. Escribir a: Jaime Escudé, Sta. Leocadia, 20. Figueres. Tel.: (972) 50 91 98.

INTERCAMBIO programas en cinta para C-64, tengo más de 800 títulos, últimas novedades. Mandad vuestra lista a: Javier Fregola. Pza. La Sal 9. 25007 Lérida.

ATENCION adictos del Commodore. Si queréis intercambiar las últimas novedades en Commodore escribir a: Joan M.ª Mas Ribes. Ctra. Castellvell, ed «Les Palmeres», esc. B atic. 3206 Reus. Tel.: (977) 31 44 05.

ESTOY interesado en comprar una unidad de disket 1541 de segunda mano, a ser posible con garantía válida y algunos programas. Posibles vendedores escribid a: Joan M.ª Mas. Ctra. Castellvell, ed. «Les Palmereses». B. 43206 Reus. Tel.: 31 44 05.

PARA recaudar fondos para la compra de nuevo equipo C-128, vendo para el C-64 lo siguiente: entre juegos y utilidades hacen un total de 250 programas, juegos como Commando, One on One, Fist I, Break Dance, The Dam Busters, etc., utilidades tales como Tasword (procesador de textos), Simon's Basic, San Reciter, etc. También vendo revistas de Commodore Worl, Magazine, Input, Tu Micro, Muy Ordenadores, Ordenador Personal, etc., y por último, vendo libros: Gráficos y sonido para el C-64, varios libros de Informática, Diccionario de Electrónica e Informática, etc. Dirigirse a: Miguel Angel Huelves García. Entre Arroyos 64, 1.º B. 20030 Madrid. Tel.: (91)4399631.

COMPRO números atrasados de la revista Tu Micro Commodore (primera época). Escribir a: José L. Pérez Hornero. Viviendas OP, 6 .(El Vivero) Trujillo 10200 (Cáceres).

VENDO Commodore 128, unidad de discos 1571 con garantía todo prácticamente nuevo con manuales y programas a escoger entre unos lona).

DESEO me envíen información acerca de la venta de juegos y utilitarios que figuran en la revista. Mi interés es saber si yo puedo comprarlos desde aquí. Escribir a: Avda. Acceso Este 2890 Nueva Mendoza (Argentina).

CAMBIO módulo expansor VIC 1020 para 6 cartuchos con dos de 8 y 16K respectivamente (para VIC 20) por cualquier material informático, electrónico o fotográfico etc. Escribir a: J. Casais. Ponte de Saa 74, Villalba 27800 (Lugo).

COMPRO unidad de discos en buen estado, también intercambio programas de todo tipo, tengo títulos tan importantes como: First I y II Ghosts'n 1000. Muy bien de precio. Llamar a José. Tel.: 332 56 90 ó 422 66 41 (Barcelona). También lo vendo por separado. Escribir: José Muñoz. Martí Codolar 47. Hospitalet (Barce-Goblins, Law of the west, Parallax, Dragon's Lair, 1942 e infinidad de juegos. Tel.: (968) 24 68 65. Escribir a: Julio Belmonte Tornel. Avda. de la Fama, 17. 30006 (Murcia).

INTERCAMBIO juegos de CBM-64 en cinta o disco. Llamar a: (93) 209 76 62. Escribir: Sergio Frías. Balmes 306, 3.º 2.ª. 08006 (Barcelona)

ESTOY interesado en el intercambio de programas para CBM-64 en cinta, poseo las últimas novedades del Reino Unido. Prometo contestar todas las cartas. Escribir a: Rogelio Ferrer Troyano. Ebro 23, (Heliopolis). 41005 (Sevilla).

ME urge mucho vender 15 interfaces copiadores para el 64. Cien por cien efectivos, los vendo sólo a gente de Madrid y alrededores a 3.000 unidad o los cambio por unidad 1541. Ofrezco 700 programas por dicha unidad. Dirigirse a: José Julio. Tel.: 895 32 54.

Kernal

No es demasiado lógico gastar el tiempo en «reinventar la rueda», sobre todo al hablar de código máquina. Sí, en muchos casos, ya existen en el ordenador los programas encargados de la tarea que intentamos realizar, ¿por qué escribirlos de nuevo?



Como ya sabemos a estas alturas, programar no es tarea sencilla (¡ni cómoda, lo cual es peor!) en muchos casos; e, incluso, puede llegar a ser casi imposible en otros.

Esto es especialmente verdad al hablar del sistema operativo de la máquina, es decir, el conjunto de rutinas que permiten controlar el teclado, la pantalla, el disco o cinta, etc. El software necesario para gestionar todas las operaciones de entrada/salida suele ser bastante complejo, o al menos, nada divertido de desarrollar.

Por tanto, una vez que el fabricante nos ha provisto de un buen conjunto de rutinas encargadas de todas estas tareas, no utilizarlas sería, cuando menos, poco inteligente. Pero existe un detalle adicional: la compatibilidad.

En efecto, un programador medianamente dotado de habilidad (y de grandes dosis de concentración y paciencia) puede desarrollar rutinas de, por ejemplo, manejo de la unidad de disco. Y tal vez llegue a crear un sistema nuevo, que funcione a 40 veces la velocidad normal, y sea toda una maravilla, y... ¡Sólo le sirva a él!

Si cada programador se dedica a su propio sistema, no será posible compatibilizar los programas de distintos fabricantes, amén de que tampoco es nada sencillo eso de hacerse un sistema operativo propio...

Menos mal que hay... ¡Kernal!

Si las razones anteriores no han logrado convencernos de lo adecuado de utilizar las rutinas propias del sistema, entonces nada lo hará (salvo, tal vez, un buen martillo... Probaremos, gracias).

En la ROM de Commodore, tenemos todo un conjunto de programas de gran utilidad: un intérprete de BASIC, rutinas matemáticas, programas de gestión de periféricos...

Utilizar una de estas rutinas es, en general, sencillo: sólo hay que conocer el punto

de entrada a la misma, es decir, la dirección en la cual comienza. A continuación, cargar los registros del procesador y las posiciones de memoria requeridas por la rutina con los valores adecuados, y efectuar el salto al punto de entrada.

Vamos a analizar en el presente artículo un grupo de rutinas de gran utilidad: las rutinas Kernal. Estas contienen todo el código de manejo de la pantalla, teclado, periféricos (disco, cinta, impresora, etc), inicialización del ordenador, proceso de interrupciones, etc. En realidad, estas rutinas son el corazón del Commodore, pues todos los demás programas las utilizan (desde el intérprete de BASIC a muchos programas de utilidades o juegos.

En términos más apropiados, diremos que estas rutinas gestionan al más bajo nivel todas las operaciones de E/S.

Entradas a las rutinas Kernal

Para comprender el sistema de acceso al Kernal, vamos a ver un pequeño ejemplo: imaginemos que la rutina que abre un fichero en disco empieza en la dirección $E000. Para emplearla, por ejemplo, podríamos hacer un JSR $E000 en nuestro programa.

En general, nada más sencillo, pero existe una posible fuente de problemas. En efecto, durante la producción de un ordenador, a veces se cambia la ROM, para corregir errores, añadir nuevas funciones, etc. Por ejemplo, esto es lo que ocurrió hace cierto tiempo en la ROM del 64, que fue modificado para corregir ciertos problemas.

¿Qué ocurriría, si durante una de esas modificaciones, la rutina de disco se desplaza de $E000 a $E001 (como suele ocurrir en las correcciones de ROM, los desplazamientos suelen ser de pocos bytes, uno en este caso)? Pues, sencillamente, que encontraríamos que nuestro programa, el cual funciona perfectamente en unas máquinas, hace las cosas más extrañas en otras, aparentemente sin más motivo que la simpatía que le inspire su dueño.

Este tipo de problema aparece, sobre todo, en máquinas más potentes, donde el sistema operativo se halla principalmente en RAM (como pueden ser los IBM PC), donde las revisiones suelen ser frecuentes o, incluso, periódicas.

Para solucionar esta fuente de problemas, se recurre a una especie de salto indirecto: en un área de memoria se mantiene una tabla con las direcciones reales de las rutinas. Para acceder a ellas, se hace el salto a la dirección de esa tabla, desde donde se relanzará a la dirección exacta, independientemente de la revisión del software en la cual nos hallemos.

Las rutinas Kernal funcionan así: En el área de memoria $FF81 a $FFF5 (65409 a 65525) se encuentran almacenadas instrucciones JMP «dirección», a razón de tres bytes por cada una. Así, un JSR $FF81 encuentra la instrucción JMP $E387 (por ejemplo), con lo cual se dirige al punto correcto. Este método de acceso asegura la compatibilidad entre todas las aplicaciones que utilicen la tabla con todas las versiones del software.

Obviamente, la descripción de las rutinas variará de una máquina a otra. Vamos a describir las rutinas del C-64 y las del C-128, salvo las especificadas de este último, las que utiliza el 64 son compatibles con otras máquinas Commodore, particularmente con el Vic 20 y el C-16.

Descripción de las rutinas Kernal

En las descripciones que siguen, vamos a analizar detalladamente cada rutina, su posición y los registros que necesita, así como las rutinas que deben ser llamadas previamente para que ésta funcione. Para ponernos de acuerdo desde el principio, vamos a emplear la siguiente nomenclatura:

FUNCION: Incluiremos el nombre oficial dado por Commodore a la rutina.

DIRECCION: La dirección a la que hemos de llamar para acceder a esta rutina (la dirección de la tabla de saltos, no la directa).

ENTRADAS/SALIDAS: Describimos los registros (tras la abreviatura REG) y/o posiciones de memoria (MEM) en las cuales debemos introducir los parámetros requeridos por la rutina o en los que la misma devolverá sus resultados. El signo «—» indicará que no se utiliza ninguno.

RUTINAS PREVIAS: En algunos casos, para que una rutina funcione es preciso haber llamado previamente a otras (por ejemplo, para enviar un carácter a la impresora tenemos que haber abierto el canal de impresora, o se producirá un error). Los nombres de las rutinas requeridas serán incluidos aquí.



POSIBLES ERRORES: Cuando a la salida de una rutina el indicador de Acarreo (C) está a uno, se indica que se ha producido un error; en este caso, el registro A contiene el código que describe el error producido. Los posibles errores se describen aquí.

STACK: Cada rutina va a ocupar una parte del stack, para almacenar variables intermedias o realizar llamadas a otras rutinas. Indicaremos la cantidad de stack que la rutina utilizará.

DESCRIPCION: De la rutina, por supuesto: efecto producido, entradas y salidas, junto a un ejemplo de su uso.

Comencemos pues con la descripción de las rutinas Kernal para el CMB-64.

FUNCION: ACPTR
Dirección: $FFA5 (65445)
Entradas: REG: —
MEM: —
Salidas: REG: A
MEM: —
Rutinas previas: TALK, TKSA
Errores: Leer status con READST.
Stack: 13

Descripción: Esta rutina recibe un byte (contenido en el acumulador) del canal serie de Commodore. Este byte viene dirigido del dispositivo que previamente ha sido inicializado mediante TALK para enviarnos datos. Ejemplo:

```
JSR ACPTR
STA DATO
JSR READST; COMPRUEBA ERRORES
```

FUNCION: CHKIN
Dirección: $FFC6 (65478)
Entradas: REG: X
MEM:—
Salidas: REG: A
MEM: —
Rutinas previas: OPEN
Errores: 3,5,6
Stack: —

Descripción: Abre un canal (siempre que deba ser otro que el teclado) como canal de entrada de datos. El número del canal a ser utilizado se incluye como parámetro de entrada en el registro X. Todo fichero que haya sido abierto previamente con la función OPEN, puede ser utilizado como canal de entrada, por supuesto, siempre que las características del dispositivo lo permitan (abrir como dispositivo de entrada una pantalla, por ejemplo, no es muy lógico que digamos...).

Esta rutina es necesaria cuando se quiera leer datos de otro dispositivo que no sea el teclado, por lo que se necesita ejecutar esta rutina antes de leer datos con otras funciones. Si no se hace, los datos serán leidos siempre del teclado, con lo que los despistes de esta clase suelen ser difíciles de encontrar.

Ahora bien, si se ha ejecutado esta rutina



y se desea volver a utilizar el teclado, se debe volver a llamarla, introduciendo en X el número del canal de teclado (0). Un ejemplo de utilización podría ser el siguiente:

```
LDX #8       ;NUMERO DEL DISCO
JSR CHKIN    ;ENTRADA DEL DISCO
BCS ERROR    ;SI C=1, A CONTIENE
              EL ERROR PRODUCIDO
.....................
LDX #0       ;NUMERO DEL TECLADO
JSR CHKIN    ;VUELVE A SER EL
              DISPOSITIVO DE ENTRADA
```

FUNCION: CHKOUT
Dirección: $FFC9 (65481)
Entradas: REG: X
MEM: —
Salidas: REG: A
MEM: —
Rutinas previas: OPEN
Errores: 3,5,7
Stack: —

Descripción: Esta rutina, muy similar a la anterior, abre un canal (siempre que deba ser otro que la pantalla) como canal de salida de datos. El número del canal a ser utilizado se introduce como entrada en el registro X. Todo fichero abierto previamente con la función OPEN puede utilizarse como canal de salida, siempre que las características del dispositivo lo permitan.

Esta rutina es necesaria cuando deseemos enviar datos a otro canal diferente de la pantalla, por lo cual es necesario ejecutar esta rutina antes de enviar dichos datos con otras rutinas. Si no se hace, los datos serán enviados a la pantalla, con lo que, al igual que en el caso anterior, los despistes de esta clase suelen ser difíciles de encontrar.

Como en el caso anterior, si se ha ejecutado esta rutina y se desea volver a utilizar la pantalla como salida, se debe volver a llamarla, introduciendo en X el número del canal de pantalla (3). Un ejemplo de utilización podría ser el siguiente:

```
LDX #8       ;NUMERO DEL DISCO
JSR CHKOUT   ;SALIDA AL DISCO
BCS ERROR    ;SI C=1, A CONTIENE
              EL ERROR PRODUCIDO
.....................
LDX #3       ;NUMERO DE PANTALLA
JSR CHKOUT   ;VUELVE A SER EL
              DISPOSITIVO DE SALIDA
```

FUNCION: CHRIN
Dirección: $FFCF (65487)
Entradas: REG: —
MEM: —
Salidas: REG: A
MEM: —
Rutinas previas: —
Errores: Leer status con READST
Stack: —

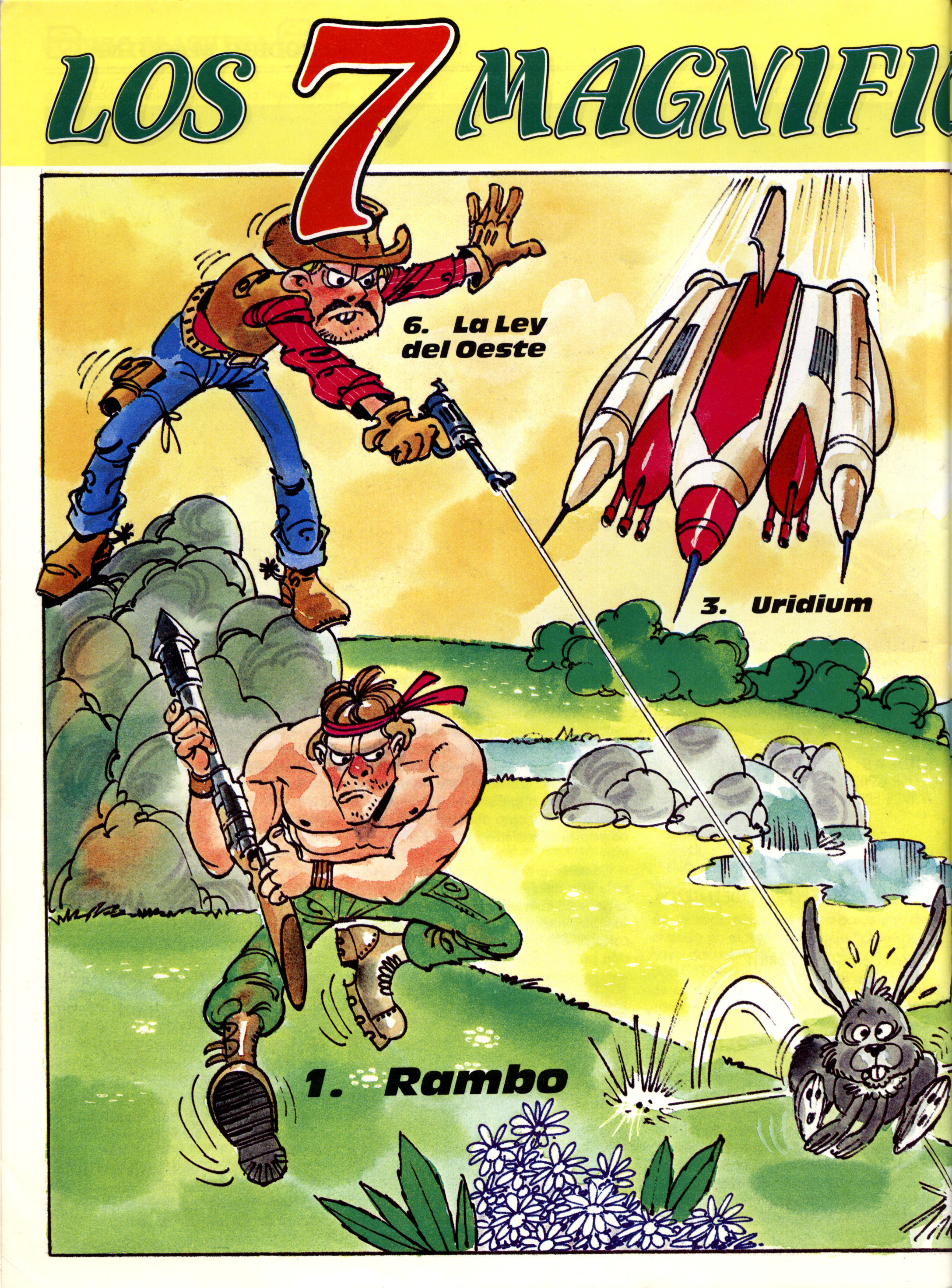
DESCRIPCION: Esta rutina permite leer un dato (el cual será devuelto en el acumulador) del canal de entrada actualmente abierto; si no se ha abierto ninguno (con la combinación de las rutinas OPEN y CHKIN) se lee del canal de datos por defecto, que es el teclado. El canal sigue abierto después de esta llamada.

En el caso particular del teclado, la primera llamada a esta rutina activa el cursor y el editor de pantalla, lo cual permite teclear los datos hasta que, finalmente, se pulsa «RETURN». En ese instante, el cursor se desactiva y la rutina devuelve el primer carácter que hayamos tecleado.

Las siguientes llamadas a la rutina van devolviendo los caracteres sucesivos, hasta que, por fin, leamos el último (que será un carácter cuyo código ASCII es 13, es decir, un retorno de carro). La próxima llamada volverá a activar el cursor, repitiéndose el proceso indefinidamente para la entrada por teclado.

Por desgracia, el espacio de que disponemos no permite más por el momento; en el próximo número continuaremos con la descripción de las rutinas Kernal.

LOS 7 MAGNIFI





Colaboremos todos en la confección de estos 7 MAGNIFICOS de TU MICRO COMMODORE. Envíanos el nombre de tu programa favorito, dejando muy claro tu nombre y dirección. Todos los meses, sortearemos cinco suscripciones por un año a nuestra revista entre las respuestas recibidas. Anímate y escríbenos a: TU MICRO COMMODORE. (7 MAGNIFICOS). Apartado de Correos 61.294. 28080 MADRID.

5. Sky Fox

7. Little Computer People

4. Comando

SUSA

1X2

2. Green Beret

BASIC 128

Aunque su nombre nos pueda engañar, este título no esconde una ampliación del ya potente BASIC del C-128; en este caso, se trata de un magnífico y completo compilador, que permite pasar a código máquina o a un código intermedio llamado P todos los programas escritos en el BASIC del C-128.

Se trata de un programa editado por Ferre Moret, con la calidad a que nos tiene acostumbrados esta casa, editora de libros para ordenadores y software de alto nivel.

Su presentación es desde luego impecable: un estuche de plástico, a semejanza de un libro y en su interior, un disco con el programa y un completo manual que ilustra las funciones del compilador con toda claridad.

Probablemente, el problema mayor del BASIC del C-128 es su, en ocasiones exasperante, lentitud, aún a pesar del comando FAST. Independientemente de esta orden, capaz de acelerar el reloj interno hasta 2 MHz, el compilador es capaz de ejecutar entre 4 y 32 veces más rápido, según las características del programa en concreto, cualquier fuente en BASIC del C-128.

Por otra parte, BASIC 128 se sale de lo habitual en el ámbito de las utilidades de su clase, dado que facilita un gran número de aplicaciones especiales que más adelante estudiaremos.



Uso del BASIC 128

Para empezar, es digno de aplauso el hecho de que el programa funcione mediante menús, lo cual permite al usuario cambiar los parámetros de compilación tales como modo FAST o SLOW, rutinas propias de coma flotante, generador de listas de líneas para hallar los errores, overlay, ampliaciones de basic, etc.

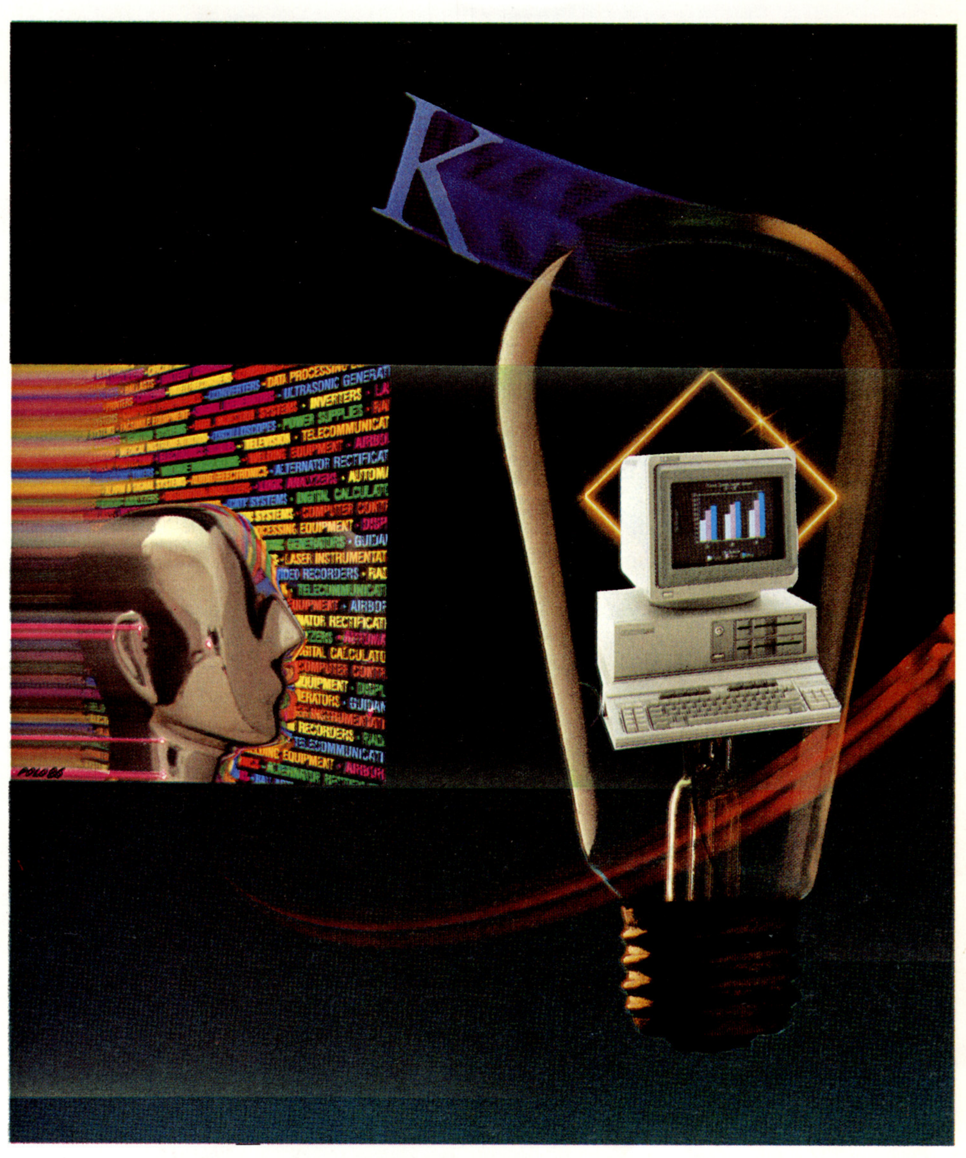
El uso del menú simplifica enormemente las cosas y el programa permite guardar los parámetros elegidos en el disco para posteriores aplicaciones.

Por otra parte, también es seleccionable el grado de compilación, o nivel de optimización de la compilación; en el primer grado (más bajo), se crea un código intermedio entre el código máquina y el BASIC, denominado código P; en el segundo nivel, se emplean variables enteras y se genera directamente código máquina.

Normalmente, el programa compilado (objeto) es más largo que el fuente en BASIC, e igualmente, el objeto es más largo que el compilado en código P; no obstante, la velocidad de ejecución es muy superior, aunque siempre depende del tipo de instrucciones utilizadas en nuestro programa fuente BASIC (bucles, tipos de variables, etc.).

En lo referente a la velocidad de compilación es bastante elevada y depende de la unidad de disco que empleemos (la 1541 ó la 1571), dado que todas las operaciones de compilación se reflejan en el disco, el cual está en funcionamiento casi todo el tiempo.

En el tiempo de compilación, el programa comprueba la sintaxis de la fuente BASIC, línea por línea, mostrándolas en la pantalla según va trabajando con cada una de ellas; si encuentra un error no se detiene la compilación, pudiéndose obtener una lista de errores para, tras corregirlos, volver a efectuar el proceso.



Aplicaciones especiales

Como aplicaciones especiales, el BASIC 128 tiene las siguientes:

A) Compilación de paquetes «overlay». Esta técnica consiste en la compilación de varios programas que usan las mismas variables, pero que por su longitud se tienen que cargar por partes en memoria.

B) Posibilidad de utilizar una serie de rutinas aritméticas de coma flotante, que permiten superar en velocidad y precisión a las implementadas en la memoria ROM del 128.

C) Capacidad para generar programas mixtos de BASIC-código P, BASIC-código máquina, y código P-código máquina, según las necesidades de memoria y optimización necesarias.

D) Capacidad de trabajar con ampliaciones del BASIC del 128.

E) Posibilidad de uso de gráficos de alta resolución en la pantalla de 80 columnas del 128.

Compilando con BASIC 128

En cuanto al manejo de BASIC 128 es bien sencillo: se carga con RUN "BASIC 128"; cuando esté en funcionamiento, aparecerá el menú de los parámetros de compilación, y una vez que los tengamos ajustados, nos pedirá el nombre del programa a compilar. A partir de este momento todo el trabajo correrá a cargo, de forma automática, del BASIC 128.

La única precaución que debemos tomar para que el BASIC 128 pueda trabajar, es disponer de bastante memoria en el disco donde se encuentra el programa a compilar, aunque no es necesario que éste sea el único presente en el mismo.

Por último, para identificar los programas y listados, el BASIC 128 utiliza los siguientes símbolos:

El programa compilado lo identifica con una P— delante y se ejecuta con RUN "P—nombre del programa".

El listado de líneas se identifica con Z—nombre del programa.

En definitiva, se trata de un programa muy útil para el programador exigente y el profesional, de manejo sencillo y con altas prestaciones, así como una buena calidad/precio.

Título: BASIC 128.
Distribuidor: Ferre Moret, S.A.
Tuset, 8, Entlo. 2.ª.
Tel.: (93) 218 02 93.
08006 Barcelona.
Precio: 5.850 ptas. (+ IVA).



CHISPA:

Cuando en el C64, queremos trabajar con un programa BASIC, unos cuantos sprites y una pantalla de alta resolución, surgen problemas de incompatibilidad de memoria.

El VIC-II al conectar el ordenador, «apunta» al banco 0 (0-16383); una pantalla de alta resolución, necesita 8K contiguas, en este caso de 0 a 8191 o de 8192 a 16383. Como en el primer caso cogemos la página cero, esa posición se descarta, quedando sólo útil la que va desde 8192 a 16383, pero esta zona está dentro del BASIC y los sprites los solemos guardar en 12288 y siguientes.

¿Qué hacer? La respuesta es subir el inicio del BASIC hasta 16384 con:POKE44,64 POKE64*256,0:NEW. Con esto, conseguimos además sitio para almacenar 32 sprites, desde 2048 hasta 4095, con lo que todo queda resuelto.

CHISPA:

En algunos programas de nuestra revista, se ha hecho uso de los relojes denominados TOD (Time Of Day). Estos no son demasiado fáciles de manejar, puesto que la representación en memoria es del tipo BCD (Decimal Codificado en Binario), lo cual significa que un número de dos dígitos, estará codificado de la siguiente forma: Las decenas ocuparán los 4 bits de mayor peso, y las unidades los 4 de menor peso. Traducido a decimal, veamos como sería, por ejemplo, la cuenta del registro de los segundos:

01, 02, 03, 04, 05, 06, 07, 08,09,16,17,18,19,20,21,22, etc.

Esto no tiene mucho sentido si no lo pasamos a hexadecimal, en cuyo caso los valores correspondientes serían:

01,02,03,04,05,06,07,08, 09,10,11,12,13,14,15,16,etc.

Las direcciones en decimal de los relojes (son dos), son:

56328: Décimas de segundo.
56329: Segundos.
56330: Minutos.
56331: Horas, bit 7 bandera AM/PM.

56584: Décimas de segundo.
56585: Segundos.
56586: Minutos.
56587: Horas, bit 7 bandera AM/PM.

Para trabajar sin problemas con ellos, conviene conocer dos peculiaridades:

1) Hay que comenzar siempre a poner en hora por el registro de las horas, y terminar por el de las décimas de segundo.

2) El reloj se detendrá al hacer un PEEK o un POKE a cualquiera de los registros de horas, minutos o segundos y continuará al acceder al de las décimas de segundo.

CHISPA:

En muchas ocasiones, es necesario presentar una salida numérica redondeada a un cierto número de decimales. Suponiendo que el número a redondear sea X, y que el de decimales deseado sea D; la función del ejemplo de la línea 10, hará el trabajo. Para utilizarla debemos definir primero en la variable D el número de decimales, y después, en la variable X, el número en cuestión. Las líneas siguientes a la 10 son un ejemplo de demostración. Veamos como puede redondear a decenas (D=—1), centenas (D=—2), etc., y no sólo a cifras decimales.

```
10 DEF FNR (X)=INT (X*10
   ↑D+5)/10↑D
20 X=148.527657: PRINT"X
   =";X: PRINT: PRINT" DE-
   CIMALES";TAB(15);"X
   REDONDEADO"
30 PRINT"---------";TAB(15);
   "------------"
40 FORD=—2TO6:PRINT-
   TAB(3);D;TAB(16);
   FNR(X):NEXT
```

CHISPA:

En todos los procesos de simulación, son necesarios los números aleatorios, los cuales son fáciles de obtener desde BASIC con la función RND; pero la cosa se complica si estamos trabajando en código máquina. Una de las mejores fuentes de que disponemos para la obtención de números aleatorios, es seleccionar la forma de onda de ruido, y leer su salida.

Para ello procederemos de la siguiente manera: colocaremos un valor de $FF en la posición $D40F, que es el valor alto de frecuencia de la voz 3, después colocamos $80 en $D418, con lo que desconectamos la voz tres, después, $81 en $D412, con lo que seleccionamos ruido en la forma de onda de la voz 3. Ahora, cada vez que necesitemos un valor aleatorio entre 0 y 255, bastará con leer el registro $D41B, que es el generador de números aleatorios de la voz 3.

CHISPA:

Recientemente, ha caído en nuestras manos una rutina en código máquina para leer los joysticks del Commodore Plus/4. El «truco», consiste en colocar el valor $FA en la dirección $FF08 si queremos leer el joystick uno y $FD (en la misma posición) si queremos leer el joystick 2.

Utilizando la llamada $A45F, que imprime un número en la pantalla, la siguiente rutina cumple el objetivo propuesto. Introdúcela con el monitor, sal de éste y teclea SYS 828, verás el número correspondiente al valor del joystick en la pantalla. Cambia el primer valor ($FA) a $FD para leer el joy #2.

```
033C LDA #$FA
033E STA $FF08
0341 LDA $FF08
0344 TAX
0345 LDA #$00
0347 JSR $A45F
034 LDA #$0D
034C JSR $FFD2
034F JMP $033C
```

CHISPA:

En el manual de la unidad de discos, existe una opción denominada SAVE-and-REPLACE (SAVE « C:nombre»), que simplemente es mejor no usar nunca. Esta opción, tiene un fallo de base en la ROM de la unidad de disco, y puede causar problemas a la larga.

Parece ser que el uso de esta opción se ve comprometido cuando el espacio disponible en disco no supera el doble del tamaño del programa a salvar. El fallo que se PUEDE producir, es un problema de BAM (Block Access Memory), y el caso más usual, es intentar cargar el programa, habiendo grabado después de efectuar la operación otro programa, y conseguir este último, habiendo perdido el anterior para siempre. Esto se puede agravar si no ponemos el 0 después del carácter @ (en apariencia funciona igual). Por lo tanto, lo mejor es no emplear este comando, y efectuar en su lugar, las siguientes operaciones, mucho más seguras.

1. Borrar el programa antiguo del disco (OPEN 15,8,15,«S:...»)

2. Grabar la nueva versión (SAVE «...»,8).

Si no queremos perder la antigua versión, bastará con renombrarla y grabar la nueva con el mismo nombre.

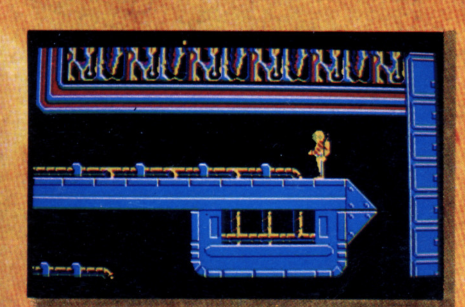
CHISPA:

Para todos aquellos que quieran saber los nombres de las personas que han fabricado su C128, el siguiente SYS es un espía perfecto: SYS32800,123,45,6.

CONQUISTAMOS EL OBJETIVO

LOS MEJORES JUEGOS AL PRECIO MAS BAJO

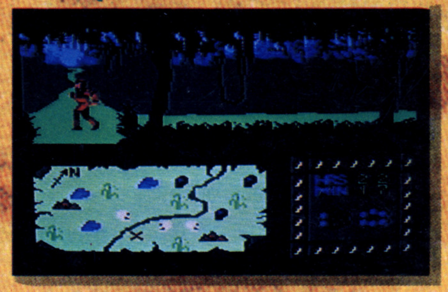




Back to Reality *COMMODORE*



Ninja *COMMODORE*



Flash Gordon *COMMODORE*

COMMODORE 64

Kane
Action Biker
Finders Keepers
Nonterraqueus
1985 The Day After
Bump Set
Kik Start
One Man And His Droid
Human Race
Speed King
ZZZ
Hollywood
Street Surfer
Super Pipeline II
Los Angeles S.W.A.T.
Back to Reality
Master Chess
Storm
Moleculeman
Xcel
Panther

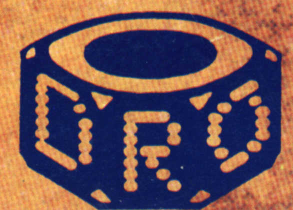
Serie M.A.D.

Five a Side
Talismán
Last V8
Ninja
Hole In One
180
Flash Gordon

SPECTRUM

Action Biker
Locomotion
Fórmula 1
Finders Keepers
Nonterraqueus
1985 The Day After

Sky Ranger
Jason Gem
Incredible Shrinking
One Man And His Droid
Space Walk
Bump Set
Storm
Pippo
Hyperbowl
Xcel
Future Games
Octagon Squad
Kane



Serie M.A.D.

Knight Tyme
Conquest
Master Of Magic
180!
La Venganza
Ninja

AMSTRAD

Locomotion
Fórmula 1
Finders Keepers
Chiller
Nonterraqueus

Oblivion
One Man And His Droid
Molecule Man
Kane
Radzone
Speed King
Storm
The Apprentice
Super Pipeline II
Back To Reality
Hyperbowl
Xcel
Hollywood or Bust
Fly Spy

Serie M.A.D.

Las V8
Knight Tyme
Five A Side
Conquest
Golden Talisman
La Venganza
180!
Ninja

M.S.X.

Fórmula 1
Finders Keepers
Chiller
Space Walk
Molecule Man
Speed King
Storm
Octagon Squad
3.2.1 Johnny Comomolo
Soul a Robot
Caves of Doom

Serie M.A.D.

Knight Tyme
La Venganza

MASTERTRONIC



DRO SOFT

DRO SOFT, FUNDADORES 3, 28028 MADRID, TELEFONO 2554500/09